1983-12

Underwater shock-induced responses of submerged cylindrical structures

Daube, Francois G.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

UNDERWATER SHOCK-INDUCED RESPONSES OF
SUBMERGED CYLINDRICAL STRUCTURES

by

Francois G. Daube

December 1983

Thesis Advisor: Y. S. Shin

Approved for public release; distribution unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Underwater Shock-induced Responses of Submerged Cylindrical Structures		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis December 1983 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Francois G. Daube		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California, 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California, 93943		12. REPORT DATE December, 1983 13. NUMBER OF PAGES 109
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS, (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Finite Elements	Underwater Explosion
Color Graphics	Submerged Structure
Fluid Structure Interaction	Nonlinear Analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The nonlinear elasto-plastic responses of a submerged cylindrical shell to an underwater shock wave have been investigated. Using the EPSA (Elasto-plastic Shell Analysis) code, the gross responses of homogeneous and ring-stiffened shells were evaluated. The relevant parameters have been displayed and evaluated using PATRAN-G color graphics system.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

An interface module was developed between EPSA and PATRAN-G. The deformations and Von-Mises stresses throughout the shell have been qualitatively evaluated.

Approved for public release; distribution unlimited.

Underwater Shock-Induced Responses of Submerged Cylindrical Structures

by

François G. Daube
Civilian , French Ministry of Foreign Affairs
M.S., Massachusetts Institute of Technology , 1982
Ingenieur E.N.S.T.A. , Paris, France, 1981

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1983

ABSTRACT

The non-linear elasto-plastic responses of a submerged cylindrical shell to an underwater shock wave have been investigated. Using the EPSA (Elasto-Plastic Shell Analysis) code, the gross responses of homogeneous and ring-stiffened shells were evaluated. The relevant parameters have been displayed and evaluated using PATRAN-G color graphics system. An interface module was developed between EPSA and PATRAN-G . The deformations and von Mises stresses throughout the shell have been qualitatively evaluated.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to express my appreciation to Professor Young S. Shin for his help and patience during this thesis and to Ray Daddazio of Weidlinger Associates for much information on EPSA.

Many thanks to Al Wong, Olive Paek and Jeanne Bowers of the Computer Science Department for their continuous support. I also want to mention the "Societe Nationale Elf-Aquitaine" (France) who financially supported this academic year.

Sincere appreciations to Lt Hugh Reams of the Defense Nuclear Agency for his support in the underwater explosion research project.

Finally, my warmest thanks to Ilaria Gorchynski who carefully reviewed the contents of this thesis and made them sound English.

I. INTRODUCTION

A. GENERAL PRESENTATION OF THE PHENOMENA

Great progress has been made in the static and dynamic analysis of complex structures through the continued development of discrete element methods of structural analysis. Tremendous improvements in computing power have made possible the study of fully nonlinear problems.

The analysis of the response of a structure submerged in a fluid, is severely complicated by the intrusion of significant fluid-structure interaction effects. Recently, the development of a variety of surface interaction approximations has provided the means for a more efficient analysis of the coupling between the structure and the surrounding fluid.

Computer codes for structural analysis are well-developed so that the fluid-structure interaction is, for the most part, handled by adding new capabilities to existing structural analysis programs.

It is a well known fact that the primary threat to a submerged structure is the shock wave that results from an underwater explosion. However, the complexity of the physical phenomena involved, along with the difficulties encountered in obtaining experimental results have been serious drawbacks for the analysis of these types of problems. But there is a definite need for investigations of large deformations and buckling problems in a structure submitted to an underwater explosion.

B. OBJECTIVES

This study deals with the nonlinear response of a submerged cylindrical shell to a shock wave. The existing finite element code EPSA (Elasto-Plastic Shell Analysis) [Ref. 1] which includes nonlinear effects and a surface interaction approximation was selected for the study. In order to alleviate the tedious interpretation of results at points throughout the shell, PATRAN-G, a color graphics system, was used. PATRAN-G allows for a global representation of a quantity distribution over the structure rather than the discrete representation given by a computer output. The objectives of this study were to merge the finite element code EPSA with the color graphics system PATRAN-G, and to conduct an analysis of the response of various types of cylindrical shells to a spherical shock wave generated by an underwater explosion.

II. THE EPSA COMPUTER PROGRAM

A. PRESENTATION

EPSA (Elasto-Plastic Shell Analysis) [Ref. 1] is a computer program developed by Weidlinger Associates and funded by DNA/NAVSEA/ONR for the purpose of the analysis of submerged stiffened shells under dynamic loadings. It incorporates a number of specific features which are geared for a more efficient analysis.

In particular, EPSA allows:
-The analysis of shells in an acoustic medium, subjected to both low and high frequency shock loadings.
-An efficient modelling of the elasto-plastic behavior
-The inclusion of large displacement effects to analyze dynamic buckling situations and post-buckling behavior.
-The modelling of stiffeners and internal structures.
-The fluid-structure interaction effect

The following sections describe the equations of motion for a submerged structural shell in an infinite fluid sujected to a pressure loading and investigate the modelling of the surrounding fluid. The last sections are devoted to the finite element discretization as well as to specific features concerning EPSA.

B. EQUATIONS OF MOTION FOR THE SHELL

Considering a thin shell of thickness h , volume V, area R submerged in an infinite fluid (figure 2.1). The shell stress resultants are defined from the stress tensor by :

$$N_{ij} = \int_{-h/2}^{h/2} \sigma_{ij}\, dt \quad \text{and} \quad M_{ij} = \int_{-h/2}^{h/2} \sigma_{ij}\, t\, dt$$

The distribution of strains (e_{11}, e_{22}, e_{12}) is assumed to be linear, the curvatures at mid-height are (k_{11}, k_{22}, k_{12}).

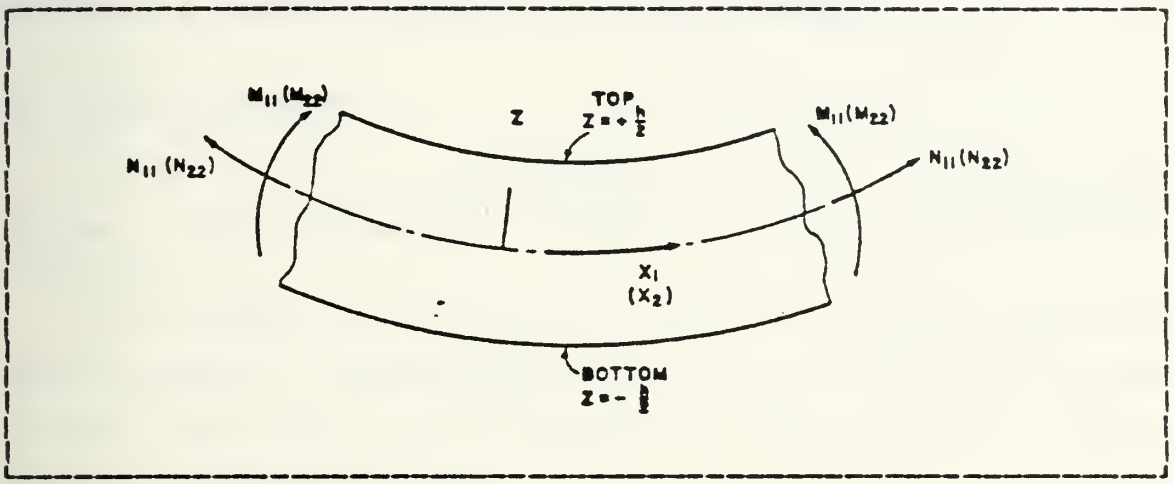


Figure 2.1 Shell Stress State.

Applying the principle of virtual work gives:

$$\int_R \{s\}^T \{\delta e\}\, dR = \int_R \{p\}^T \{\delta u\}\, dR - \int_R \rho \{u\}^T \{\delta u\}\, dR \qquad (2.1)$$

Where

$\{u\} = (u_1, u_2, w)^T$ is the displacement vector at each point

$\{s\} = (N_{11}, N_{22}, N_{12}, M_{11}, M_{22}, M_{12})^T$ is the stress resultant vector

$\{e\} = (e_{11}, e_{22}, 2e_{12}, k_{11}, k_{22}, 2k_{12})^T$ is the strain resultant vector

ρ is the mass per unit area for the shell.

The symbol δ will refer to a virtual quantity, and the dot or star denotes a differentiation with respect to time.

The first term of equation (2.1) represents the virtual work of internal forces , the second represents the virtual work of external forces (i.e. pressure, point loading, etc), and the third one represents the contribution of inertia forces in the virtual work. Thus, this last term expresses the effects of dynamic phenomena on the structure.

C. FLUID MODELING

In the case of a submerged structure, the external forces are the pressures applied at the fluid-structure interface.

As the shock wave hits the stucture it gets reflected, thus inducing a pressure term p_r . In addition, the motions of the shell will also generate a radiated wave, with a pressure contribution p_{ra}
Therefore, the pressure at the fluid structure interface is the sum of the incident, reflected and radiated pressures:

$$p = p_i + p_r + p_{ra} \qquad (2.2)$$

Where
p = Total dynamic pressure.
p_i = Pressure associated with incident free field pressure wave.
p_r = Reflected pressure due to the interaction of the incident wave with the structure, the structure being fixed and rigid.
p_{ra}= Radiated pressure due to the normal movements of the surface of the structure in the fluid
p_s= p_r+ $p_{ra}$is called the scattered pressure.

The methods for getting the scattered pressure p_s will now be investigated. Assuming a spherical wave in an

acoustic medium with a sound speed c , the pressure is determined by the well known wave equation:

$$c^2 \nabla^2 p = \frac{\partial^2 p}{\partial t^2} \qquad (2.3)$$

and the proper conditions at the boundaries of the fluid domain .

One alternative for solving the previous problem would be simply to use a finite element discretization of part of the fluid domain , imposing a radiation condition at the boundary [Ref. 2]. However, this would require a very significant fraction of the computing effort that could not be devoted to the structural modelling.
Therefore, a more efficient way of computing the scattered pressure is required.

The Doubly Asymptotic Approximation (DAA) imparts upon the structural model a surface loading composed of incident and scattered waves.

In the high frequency limit, it can be shown that the scattered nodal force vector $\{F_s\}$ is related to the wave particle velocities normal to the structure's surface by [Ref. 3] :

$$\{F_s\} = [A]\{U_s\} \qquad (2.4)$$

Where $[A]$ is the diagonal matrix of nodal areas (areas associated with each node) and $\{U_s\}$ is the vector of nodal scattered normal velocities. Therefore, in this high frequency case the shock wave is simply a plane wave and equation (2.4) states that the pressure is proportional to the fluid velocity.

In the low frequency limit the fluid structure interaction is governed by the relation:

$$\{F_s\} = [M_v]\{U_s^*\} \tag{2.5}$$

Where $\{U_s^*\} = \frac{d}{dt}\{U_s\}$ is the nodal normal acceleration vector and $[M_v]$ is the added mass matrix computed in an incompressible fluid.

Thus, in the low frequency case, the loading is due to the motion of the rigid structure in an incompressible fluid, a problem typically found in hydrodynamics.

When a broader range of frequencies is considered, one can combine the two previous equations with the differential equation governing the scattered pressure [Ref. 3], giving:

$$[A]^{-1}\{F_s^*\} + \rho c[M_v]^{-1}\{F_s\} = \rho c\{U_s^*\} \tag{2.6}$$

Where $\{F_s^*\} = \frac{d}{dt}\{F_s\}$

Defining the vector of nodal scattered pressures $\{p_s\}$ by:

$$\{p_s\} = [A]^{-1}\{F_s\}$$

we get:

$$[M_v]\{p_s\} + \rho c[A]\{p_s\} = \rho c[M_v]\{U_s^*\} \tag{2.7}$$

which is the set of equations that gives the scattered pressure at each node of the wetted surface of the shell.
Equation (2.7) gives exact results in both the high and low frequency limits. Thus, DAA allows the modelling of the fluid-structure interaction via a coupled set of differential equations at the wetted nodes of the structure instead of modelling the whole fluid with a finite element grid. The use of DAA will free some memory space in the computer

for a better modelling of structural behavior while at the same time giving some reasonably accurate results for the loading of the structure.

Therefore, the equations for the study of underwater shocks will consist of a coupled set of structural equations that come from the Principle of Virtual Work and of fluid equations at the wetted nodes that come from DAA equations.

D. FINITE ELEMENT PROCEDURE

1. Discretization

The principle of virtual work is rewritten for the structure introduced in section B [Ref. 4]

$$\int_R \{s\}^T \{\delta e\}\, dR - \int_R \{p\}^T \{\delta u\}\, dR + \int_R \rho \{\ddot{u}\}^T \{\delta u\}\, dR = 0 \qquad (2.8)$$

The surface of the region is covered by a quadrilateral mesh of N elements, each having an area A_i . The previous integral can then be replaced by a summation of integrals over A_i .:

$$\int_R \{s\}^T \{\delta e\}\, dR = \sum_{i=1}^{N} \int_{A_i} \{s\}_i^T \{\delta e\}_i\, dR \qquad (2.9)$$

The integrations over A_i are then performed by dividing the element into four regions A_i^k (figure 2.2) We have then:

$$\int_{A_i} \{s\}_i^T \{\delta e\}_i\, dR = \sum_{k=1}^{4} \{s\}_i^T \{\delta e_k\}_i A_i^k \qquad (2.10)$$

and therefore equation (2.9) becomes :

$$\int_R \{s\}^T \{\delta e\}\, dR = \sum_{i=1}^{N} \sum_{k=1}^{4} \{s\}_i^T \{\delta e_k\}_i A_i^k \qquad (2.11)$$

Using the same procedure, it can also be derived :

$$\int_R \{p\}^T \{\delta u\}\, dR = \sum_{i=1}^{N} \sum_{k=1}^{4} \{p\}^T \{\delta u\}_i A_i^k = \{P\}^T \{\delta q\} \qquad (2.12)$$

$$\int_R \rho \ddot{\{u\}}^T \{\delta u\}\, dR = \sum_{i=1}^{N} \sum_{k=1}^{4} \ddot{\{u\}}_i^T \{\delta u\}_i A_i^k = [M] \ddot{\{q\}} \{\delta q\} \qquad (2.13)$$

Where [M] is the mass matrix , {P} is the vector of externally applied forces, and {q} is the vector of nodal normal displacements for the structure.

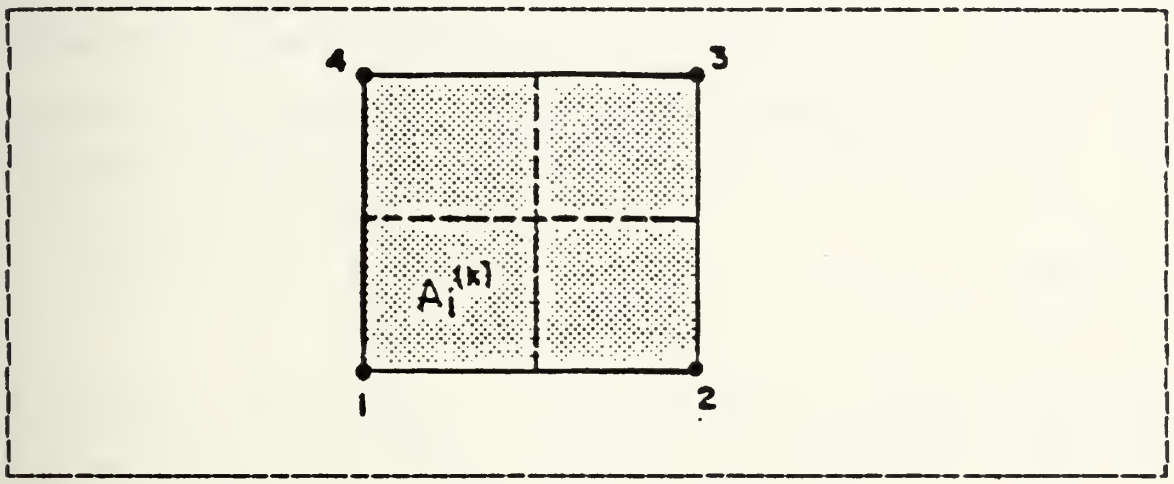


Figure 2.2 Grid Points in EPSA.

By definition, finite element discretization can express the displacement {u} at any point of an element as a function of the displacements at the corner points of the element, defined by {q} .

$$\{u\} = [H]\{q\} \qquad (2.14)$$

Where [H] is a 6×12 matrix of interpolation functions.

Combining the derivatives of {u} will give the strain vector {e}. In matrix form, equation (2.14) gives :

$$\{e\} = [B]\{q\} \qquad (2.15)$$

Where [B] is a 6×12 matrix function of the geometry of the element as well as a function of the previous deformations in the case of large displacements.
Using the previous result, equation (2.9) is rewritten in the following way :

$$\int_R \{s\}^T \{\delta e\}\, dR = \sum_{i=1}^{N} \sum_{k=1}^{4} \{s\}_i^T [B_k]_i \{\delta q\}_i A_i^k = \{F\}^T \{\delta q\} \qquad (2.16)$$

Where {F} is the internal force vector.

Combining the previous equations in equation (2.8) , the principle of virtual work becomes:

$$[M]\{\ddot{q}\} = \sum_{i=1}^{N} (\{P\}_i - \{F\}_i) \qquad (2.17)$$

Therefore, the principle of virtual work has been tranformed into a system of ordinary differential equations which are much more amenable to numerical treatment.

In EPSA, each arbitrarily shaped quadrilateral element is defined by four corner nodes, each having three translational and no rotational degree of freedom. In order to represent the behavior in bending eight nodes not contiguous with the element are also used (figure 2.3). Every element accesses twelve nodes and has twenty degrees of freedom: three translational degrees at the corner nodes and one corresponding to the displacement normal to the surface for each of the eight exterior nodes. The bending behavior (second derivative term) is expressed in terms of the nodal displacements via a finite difference technique [Ref. 4].

Then the nodal displacement vector of element i is simply:

$$\{q\} = (u_1, u_2, u_3, u_4, v_1, \ldots v_4, w_1, \ldots w_4, w_5, \ldots w_{12})^T \quad (2.18)$$

Conventional finite element codes utilize three translational and two rotational degrees of freedom at each node (each element has 5 4=20 d.o.f. as in EPSA). However the masses associated with rotational degrees of freedom are very small, leading to numerical instability. The use of the aforementioned formulation alleviates this problem since only translations are considered and, in addition, the number of unknowns is reduced, leading to simpler and more efficient computations.

It must be pointed out that in order to model the bending behavior at the edges of the shell, a set of artificial nodes has to be created. The finite element grid will then consist of the nodes defined in the input file plus the artificial nodes along the boundary of the sheet (figure 2.3).

2. Strain Displacement Relation

The principle of virtual work deals only with strains. Since the finite element approximation gives the displacement at each point, the equations relating the strains to the displacements are needed. In this study, the Donnell-Vlasov nonlinear kinematic equations of shell theory are employed, and the strain-displacement relations are described in table I.

Using equation (2.15) in the equations detailed in the previous section will give the finite element approximation of strains in terms of nodal displacements.

TABLE I

Donnell-Vlasov Shell Equations

$$e_{11} = \frac{\partial u_1}{h_1 \partial \xi_1} + \frac{1}{h_1 h_2} \frac{\partial h_1}{\partial \xi_2} u_2 + \frac{w}{R_1} + \underline{\frac{1}{2} \phi_1^2}$$

$$e_{22} = \frac{\partial u_2}{h_2 \partial \xi_2} + \frac{1}{h_1 h_2} \frac{\partial h_2}{\partial \xi_1} u_1 + \frac{w}{R_2} + \underline{\frac{1}{2} \phi_2^2}$$

$$2e_{12} = \frac{\partial u_2}{h_1 \partial \xi_1} + \frac{\partial u_1}{h_2 d\xi_2} - \frac{1}{h_1 h_2} \frac{\partial h_1}{\partial \xi_2} u_1 - \frac{1}{h_1 h_2} \frac{\partial h_2}{\partial \xi_1} u_2 + \underline{\phi_1 \phi_2}$$

$$k_{11} = \frac{\partial \phi_1}{h_1 \partial \xi_1} + \frac{1}{h_1 h_2} \frac{\partial h_2}{\partial \xi_1} \phi_2 \quad k_{22} = \frac{\partial \phi_2}{h_2 \partial \xi_2} + \frac{1}{h_1 h_2} \frac{\partial h_2}{\partial \xi_2} \phi_1$$

$$2k_{12} = \frac{\partial \phi_2}{h_1 \partial \xi_1} + \frac{\partial \phi_1}{h_2 \partial \xi_2} - \frac{1}{h_1 h_2} \frac{\partial h_1}{\partial \xi_2} \phi_1 - \frac{1}{h_1 h_2} \frac{\partial h_2}{\partial \xi_1} \phi_2$$

Where $\phi_1 = - \frac{\partial w}{h_1 \partial \xi_1}$ $\phi_2 = - \frac{\partial w}{h_2 \partial \xi_2}$

ξ_1, ξ_2 are local coordinates

h_1, h_2 are scaling coefficients.

3. Shell Constitutive Relations

The shell constitutive relations relate the stress resultant rate vector to the shell strain rate vector. In matrix terms:

$$\{s\} = [D]\{e\} \qquad (2.19)$$

Where [D] is the Elasto-Plastic tangent stiffness matrix.

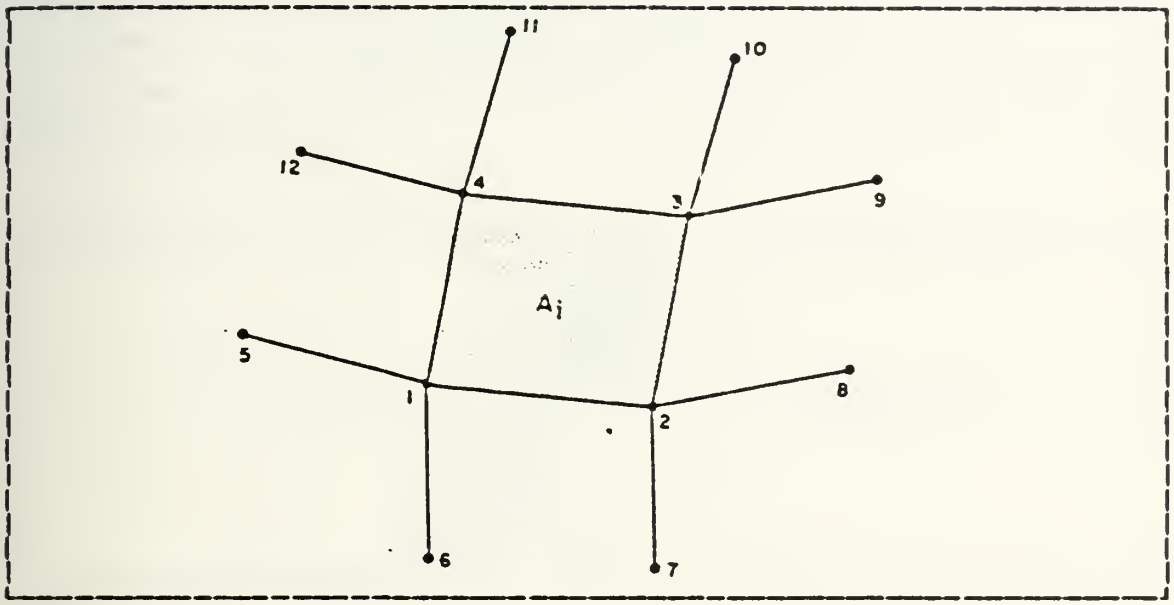


Figure 2.3 Nodal Points Organization.

The stress-strain relation used in EPSA differs from the classical Elasto-Plastic theory in that the formulation involves shell stress resultants rather than stresses at points throughout the thickness of the shell. In other words, EPSA uses relation 2.19 integrated over the thickness of the shell. Thus, there is no need to compute the stresses throughout the shell, which results in a significant savings in storage space and processing time. However, the stress resultants N_{ij} and M_{ij} of the shell theory are not sufficient to describe the state of stress, and certain higher order moments must be combined with the stress resultants to form the dynamic variables of the problem. The coefficients for the integrated constitutive equation have been determined using experimental results [Ref. 5].

These constitutive equations consist of a yield condition, a strain hardening law and a flow rule:
-The yield condition indicates whether part of the shell has started to yield. (figure 2.4)
-The strain hardening law gives the evolution of stresses in the shell after plasticity is reached.
-The flow rule gives the plastic strain rate in the shell after plasticity.

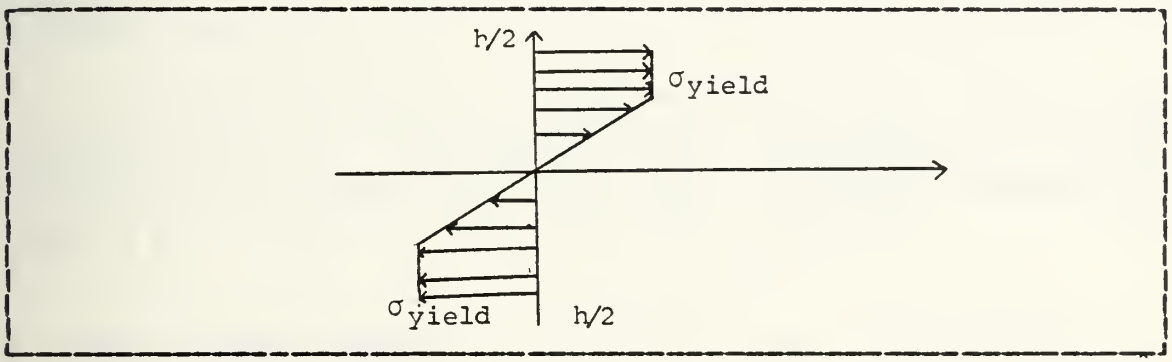


Figure 2.4 Yield Situation in the Shell.

4. Solution Procedure

EPSA uses an explicit central difference scheme to solve the virtual work and fluid loading equations detailed in section B. As indicated in appendix B, the explicit time integration procedure requires a small time step and is only conditionally stable. However when stable, it always converges to the exact solution, as opposed to implicit schemes that are unconditionally stable but may lead to unrealistic results. Furthermore,in problems involving the treatment of shocks, accuracy requirements preclude the use of large time steps. The central difference scheme seems, therefore, particularly optimal.

Assuming we know the solution at time t ,the central difference scheme applied to equation (2.17) gives the solution at time t+Δt :

$$\{V\}_i^{t+\Delta t} = \{V\}_i^{t} + \frac{\Delta t}{M_i} \left(\{P\}_i - \sum_{k=1}^{4} \{F\}_k \right) \qquad (2.20)$$

Where M_i is the mass of node i , $\{P\}_i$ are the externally appied forces and $\{F\}_k$ are summed over all the elements k framing node i.

The formulation of the equations is in the initial configuration and all equations are solved in the initial geometry in accordance with the total Lagrangian formulation [Ref. 6].

E. EPSA CAPABILITIES

The structure to be analyzed is conceptually divided into constitutive parts called "sheets." Each sheet is a curved section of a shell with an arbitrary number of nodes and elements (figure 2.5) Its shape is limited to a surface that can be described by a smooth continuous function without discontinuities in its slope. The elements within the sheet are limited to a rectangular organisation (figure 2.5).

Thus, a cylinder with end caps would consist of three sheets: a cylindrical sheet and a sheet for each end (figure 2.5). Three sheets are required because of the slope discontinuity at the edge between the cylinder and the end caps.

Dividing the structure into sheets isolates the difficulties associated with the boundaries into a set of artificial nodes along the perimeter of the sheet. When several sheets are required ,EPSA takes care of the compatibilities of displacements, rotations and moments along the edges.

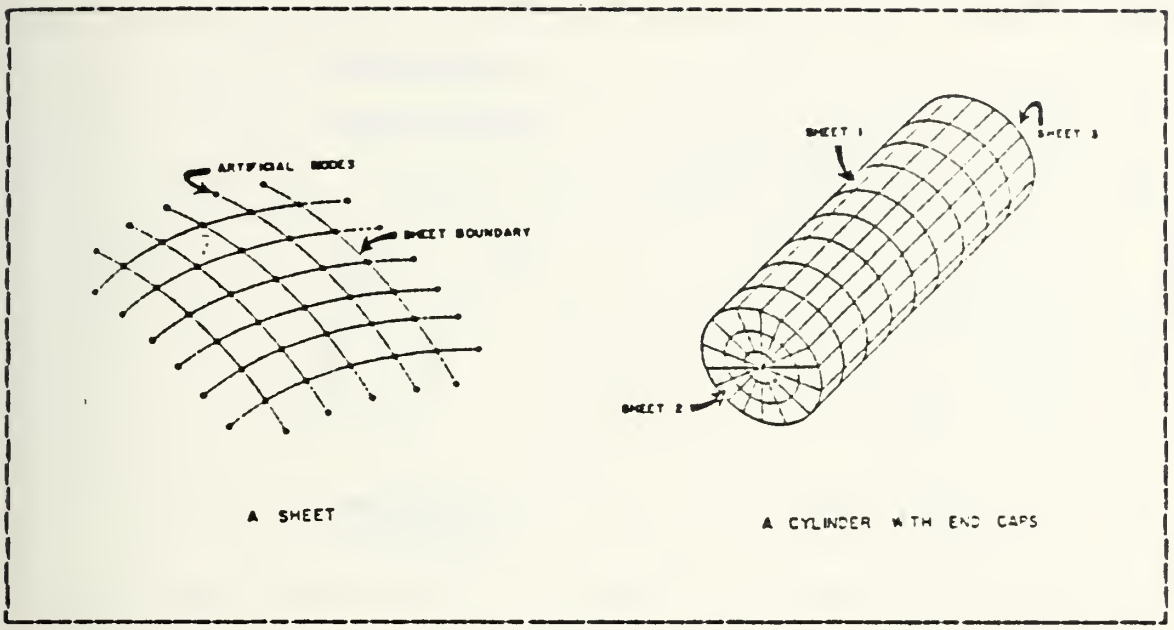


Figure 2.5 Sheet Organization.

Thus, any arbitrarily-shaped structure can be analyzed using EPSA by dividing it into a number of sheets.

Two options for choosing elements are available in EPSA. The first option exists to employ a generalized quadrilateral element. The second option exists to employ a rectangular element and uses a specialized formulation for this type of element.

The coordinates which can be either cartesian or cylindrical always lie within the plane of the sheet. The z direction lies in a positively outward direction normal to the sheet. Each sheet contains its own local coordinate system, there is no global coordinate system when multi sheets are merged (figure 2.6).

Prior to generating a finite element mesh for a sheet one must establish the side numbers of the sheet. The side numbering scheme is arbitrary as to the choice of sheet number one. However the specification of sides 1 to 4 must proceed in a counterclockwise direction when the sheet is

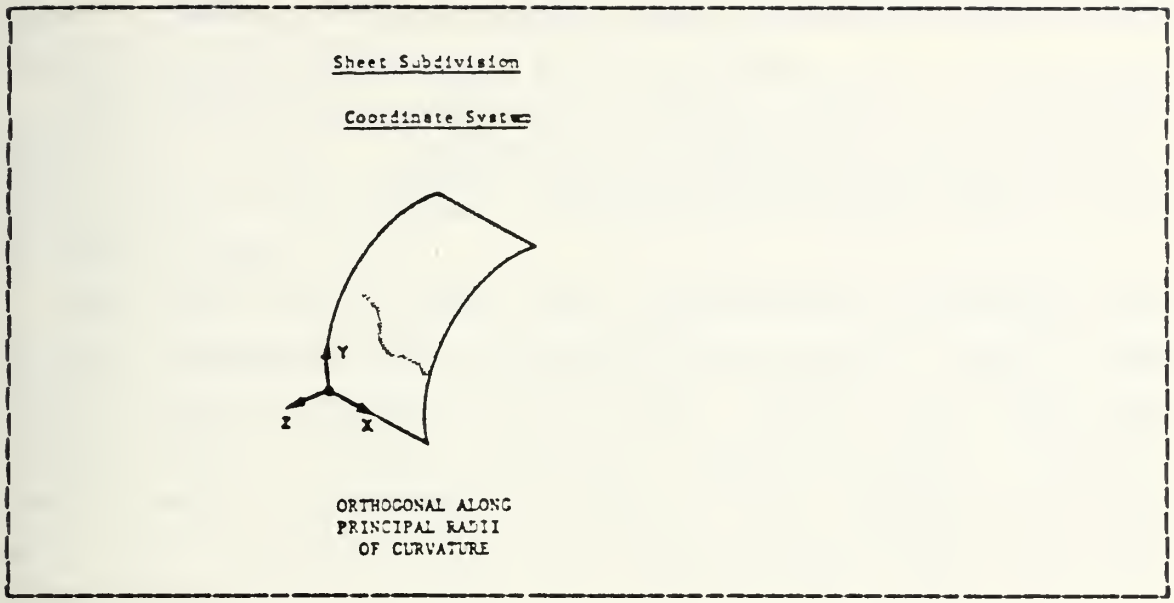


Figure 2.6 Coordinate System.

viewed from the positive z direction. At the intersection of side 1 and side 4, element 1 is first defined. Then the node/element number is incremented by one until it reaches side 2. Then it returns to side 4 and continues the count for the next row of nodes/elements.

Thanks to the exclusive use of quadrilateral elements and to the specific numbering procedure, the table of connectivity is implicitly defined when generating the nodal points, thus simplifying considerably the input requirements.

The inclusion of structures internal to the cylindrical sheet is enacted in EPSA through the use of internal sheets. Structures internal to a cylinder are therefore modelled as individual sheets having the same capabilities as any general EPSA sheet.

Two types of internal structures are available:
-Sheets (beams, plates or shells) whose connection to the cylindrical shell lies parallel to the axis of the cylinder.

-Sheets (beams, plates, shells) whose connection lies in the circumferential direction of the cylinder.

Furthermore, in order to model internal equipment (machinery, etc) which does not deform but contributes to the inertia of the system, concentrated masses can be input at nodal points.

The user must be aware that the previously discussed DAA is only implemented for cylindrical stuctures. Prior to the finite element analysis the user must compute the added mass (virtual mass) matrix defined in equation (2.4). This is done by using the ACCESION program, which creates a virtual mass file that EPSA reads when computing the fluid-structure interaction.

Finally, EPSA in its latest version takes local cavitation into account. When the total pressure computed by EPSA is found to be negative, it is simply set to zero since water cannot withstand any tension.

F. USING EPSA

The input file for EPSA can be generated either directly, or via the interactive program PREPSA that prompts the user to give the input data . When the input file is created, all the data are in free format.

The nodal points can be generated semi-automatically (see the user's manual), and this option is very helpful and time saving when generating big models. EPSA can be run interactively for small models or on batch for bigger jobs. For instance, a cylindrical shell with 1440 elements and 1517 nodal points takes 0.0129 sec. of VAX CPU per time step per element. The whole model would take about half an hour for 100 steps.

EPSA creates an output file in which all input data is echoed, and outputs the pressures, stresses, strains, veloc-

ities, displacements at the nodal points requested by the user in the input deck.

The value of the time increment Δt should be selected so that :

$$\Delta t \leq 1/2\, \delta_{min} (\rho/E)^{\frac{1}{2}} \qquad (2.21)$$

Where δ_{min} is the smallest distance between the midpoints of opposite sides of an element, for all elements of the sheet (figure 2.5), and 1/2 is a safety factor. In other words, the time step increment has to be less than the time taken by a wave to propagate from an element to another. In equation 2.21, $(E/\rho)^{\frac{1}{2}}$ is simply the wave speed in the material.

The virtual mass array (VMA) is created on unit 20, therefore one should not use this unit for any other purpose than READ operations.

Because of the simple organization of its input file, EPSA has been found fairly easy to use. The user can perform major changes in the model very quickly and efficiently. The ACESION module has been found to work well except for cylinders of large dimensions (L=1400", D=240") for which the size of the virtual mass array grows unexpectedely from a reasonable 4 blocks to 190 blocks of VAX memory size.

However, EPSA has been designed for some specific types of fluid-stucture interaction analysis and its limitations should be pointed out:

- Only beam type stiffeners can be considered
- The fluid structure interaction is only enacted for a circular cylindrical sheet, which takes away much of the flexibility the program.

III. EPSA/ PATRAN-G INTERFACE

A. INTRODUCTION TO COLOR GRAPHICS SYSTEMS

In dealing with the response of a structure to a loading, quantities such as stresses, strains, velocities and displacements must be analyzed at a number of nodal points, which makes the interpretation of computer outputs very tedious. Color graphics systems offer an effective solution to this problem by providing a global representation of a quantity distribution over a stucture rather than a discrete representation given by a computer output. A color graphics system consists of an interface between the computer, the user and the color terminal. A typical system would be a software package that allows the user to create models on the screen as well as to display any data in terms of color intensity. It is known that a picture is worth several hundreds of words. Therefore, merging a finite element program with a color graphics system would be a major improvement in engineering analysis.

PATRAN-G [Ref. 7] is a color graphics system specifically designed for finite elements, it permits the engineer and the computer to work together towards the creation of a model. The designer creates an image on the screen and PATRAN automatically translates the physical model into a standard finite element input deck. Another important feature of PATRAN-G is its ability to assist the user in the interpretation of results through its post-processing facilities which include: the deformed geometry with magnified deformations,the color coding of elements based upon any response parameters such as displacements, stresses and strains. In particular, the contour levels of the

aforementioned quantities can be superimposed on a 3-dimensional image of the model, thus allowing for a global analysis of a complex structure.

B. MERGING EPSA AND PATRAN-G

As described in chapter I , the structures under study have fairly short EPSA input files. Furthermore, in this study dealing with fluid-structure interactions on a circular cylinder, only structures that consist of one sheet are considered. Therefore, because of the simplicity of both the input file and the stucture under study, there was no need to design a module converting a PATRAN-G model that is created on the screen into an EPSA input file.
The remaining tasks were the following:

-Display the original finite element model defined in the EPSA input file on the screen (original geometry).

-Display the nodal points and element results that are computed from EPSA on the screen (postprocessing).

1. Original Geometry

The input of a finite element model into PATRAN-G can be done by creating a "neutral" file, as described in PATRAN-G terminology. The neutral file[1] is intended to provide a simple link between PATRAN-G and the outside world. It is written entirely in 80 character card images and all the data is organized in small "packets" of two or more card images. Each packet contains the data for a fundamental unit of the model such as node coordinates or elements definition. Since our only purpose was to display the original geometry of the structure, a limited number of

---

[1]Additional information about neutral files can be found in the PATRAN-G user's manual [Ref. 7]

data packets (4) was to be created:

-File title (packet 25) : two cards containing the user title.

-Summary data (packet 26) : two cards containing the number of nodes, elements and the date and time at which the neutral file was created.

-Node data (packet 1): contains all information concerning nodes: node number and coordinates in a global coordinate frame.

-Element data (packet 2) : contains the connectivity table for the finite element model.

-End of file (packet 99) : end-of-file cards.

We have seen in the presentation of EPSA in chapter I that the nodes are defined in a local, sheet-attached coordinate system, that artificial nodes are created along the edges of the sheet to model the bending behavior, and that no connectivity table was input. Therefore, the translator module that would be created had to:
-skip the set of artificial nodes and properly renumber the grid
-perform a change of coordinates for all local data
-generate the connectivity table.

It was decided to employ a modular design in which each routine would perform a specific task. A modular design would allow further changes to be made quickly and efficiently by modifying only the relevant routines. The implementation in EPSA was made using a series of "calls", thus minimizing the risk of interference with the finite element computations.

The translator module created was made of four routines:

-PRELIM : computes the number of nodal points, the number of elements and displays the first two data packets (25,26)

-SHEETF : scans through the nodes, skips artificial nodes, renumbers the grid, performs the required changes of coordinates and displays the node data packet (01)

-SHCONN : scans through the nodes, connecting each node to the elements it belongs to and displays the element data packet (02) on the neutral file.

-ENDNEU : displays the end-of-file packet (99)

2. Using the Translator Module

The translator calls were implemented in the routine REPORT of EPSA. Any run of EPSA creates a neutral file on unit 19. The neutral file name is therefore FOR019.DAT if no "ASSIGN" statement has been issued prior to the computer run. The finite element model might then be displayed on the graphic terminal (Ramtek, Tektronix) via the neutral input mode of EPSA (see [Ref. 7] for more details). An example of finite element model output on Tektronix 4014 is given on figure 3.1 .

3. Implementation of Postprocessing Capabilities

Postprocessing capabilities exist to assist the user in the interpretation of computer results. It is simply a process of generating displays and reports based upon a combination of geometry and the results of an analysis.

The results of analyses are transmitted to PATRAN-G for postprocessing via "neutral results files" as decribed in PATRAN-G terminology. Unlike the model neutral file described in the previous section, results files are in binary rather than in card image form.

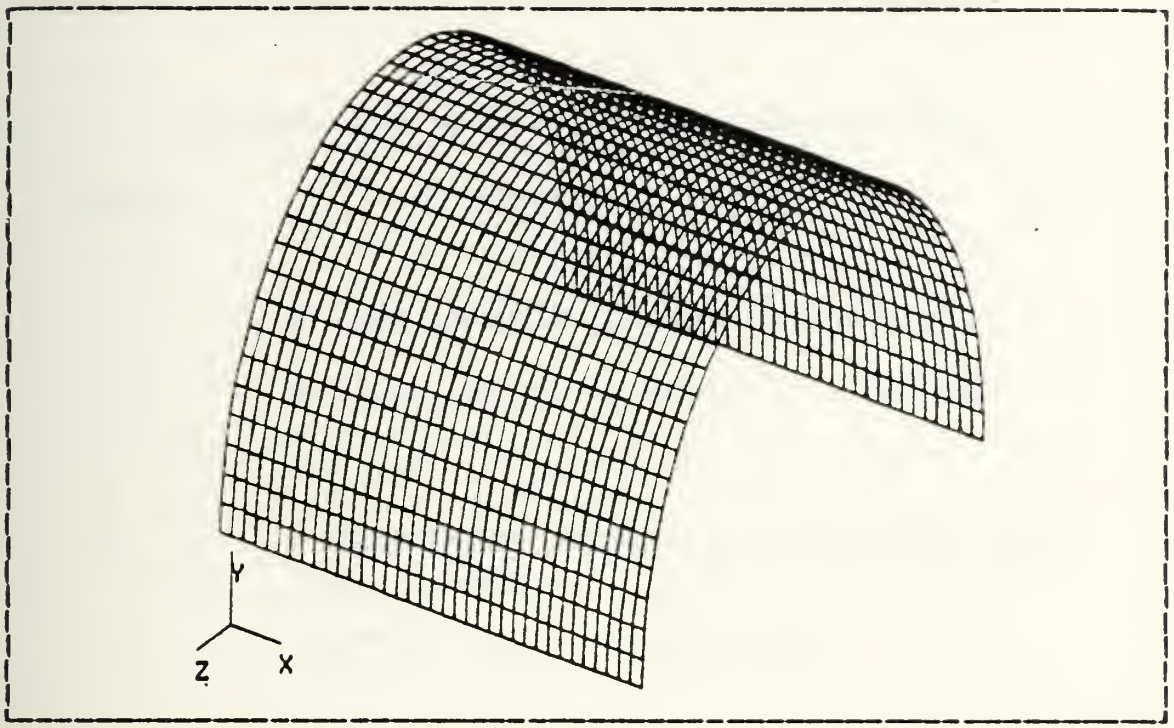


Figure 3.1 Example of Finite Element Model Display.

One can distinguish between two major kinds of post-processing displays: deformed geometries and element quantities.

a. Deformed Geometry

A displacement results data file required by PATRAN-G had to be created. Again, the module created had to skip the artificial nodes, perform changes of coordinates as well as to write the nodal diplacements in the neutral results file. The displacement results data file was created in module NEUDISP. Its organization is given on table II.

A small module PLOTDISP that decides at which time steps the results should be output was created. The "call" to PLOTDISP was implemented in module COMPUTE of EPSA. The overall structure of the translator module is presented on table IV at the end of the chapter.

TABLE II

Organisation of Displacement Results File

Column	Content
1	X displacement in global coordinates
2	Y displacement in global coordinates
3	Z " " "
4	Displacement normal to the shell without rigid body mode.
5	Velocity normal to the shell

A sample of deformed geometry output on Tektronix 4014 is given in figure 3.2.

b. Element Quantities Postprocessing

Any element related quantity can be the target of postprocessing. In general these types of quantities are the results of finite element computations supplied to PATRAN-G through the neutral element results file. The neutral element results file is different from the neutral displacements results file detailed in the previous section, however it shares a similar format [Ref. 7]. Element quantities such as stresses in local and global coordinates and von Mises stresses are computed in module NEUSTRE whose overall structure is similar to NEUDISP described earlier.

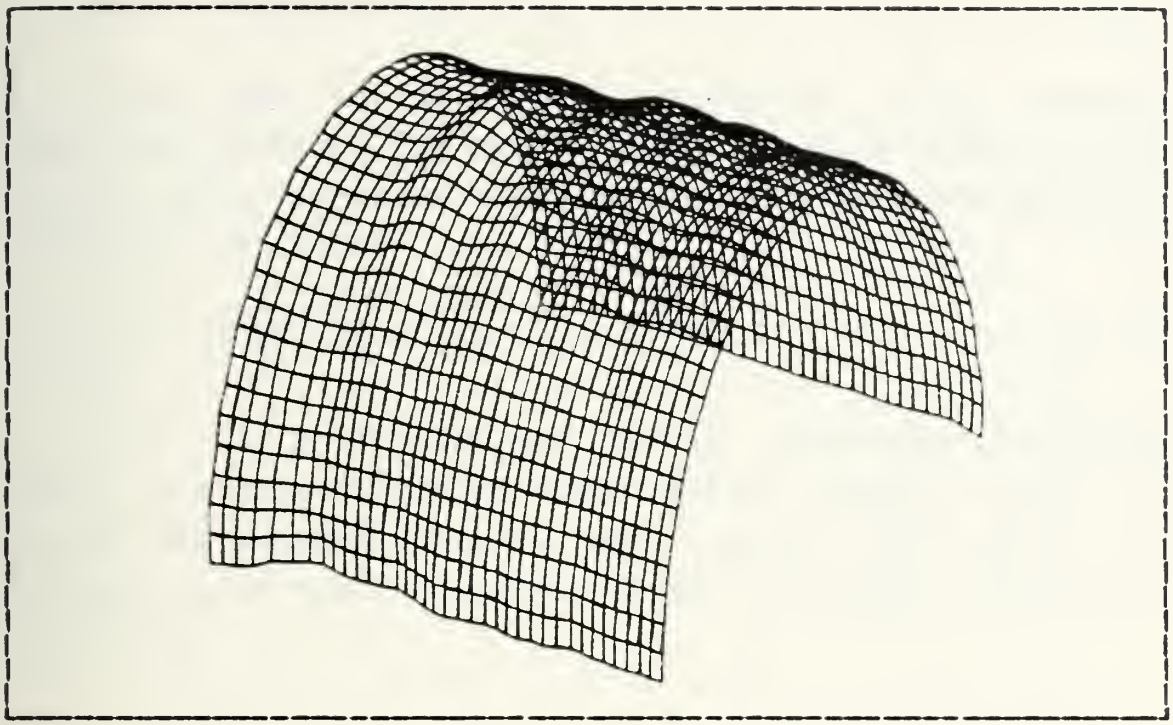


Figure 3.2 Deformed Geometry Output.

The organisation of the neutral element results file is given on table III.

As described in chapter I, EPSA does not compute the stresses through the thickness of the shell. Instead one uses the integrated quantities of the shell theory [Ref. 8]

$$N_{ij}=\int_{-h/2}^{h/2}\sigma_{ij}\,dt \quad \text{and} \quad M_{ij}=\int_{-h/2}^{h/2}\sigma_{ij}\,t\,dt$$

One can expect the stresses on the shell to be maximum at the extreme fibers. NEUSTRE computes the von Mises stresses at the top and bottom fibers and writes the maximum value in the neutral file. At the surface of the shell no shear stresses are involved. Assuming a linear distribution of bending stresses and a uniform distibution of membrane stresses, one can easily derive :

$$\sigma_{ij} = N_{ij}/h + M_{ij}\,6/h^2 \qquad (3.1)$$

The first term of the previous equation is the membrane force contribution, the second is the bending moment contribution. The von Mises stresses are then computed using the well-known relation:

$$\sigma_{vm} = \sqrt{(\sigma_1^2 - \sigma_1\,\sigma_2 + \sigma_2^2)} \qquad (3.2)$$

In a similar way to the displacements results file, a module PLOTSTRE that decides whether or not to output the element results was created and called from COMPUTE. The overall structure of the translator is given on table IV.

TABLE III

Organisation of the Neutral Element Results File

Column	Content	Description
22	stre,1	Element local stress, direction i
23	stre,2	" " " " j
25	stre,4	Element global stress, direction x
26	stre,5	" " " " y
27	stre,6	Element global shear, direction xy
31	von	von Mises stress

c. Displaying the Results Quantities on the Screen.

The EPSA input deck has been modified so as to create the results files (displacements, elements) required by PATRAN-G. At the end of the second input card the user specifies the number of displacement results and the number of element results files to be created (at least one). Obviously, those two inputs are also in free format. The neutral results files (elements, displacements) will be generated at equal time intervals as the computation proceeds. The results corresponding to the last time step are always output.

The element results file is created on unit 16 and the displacement results file on unit 18, thus corresponding to files FOR016.DAT and FOR018.DAT respectively. A new version of those files is created each time an output is requested.

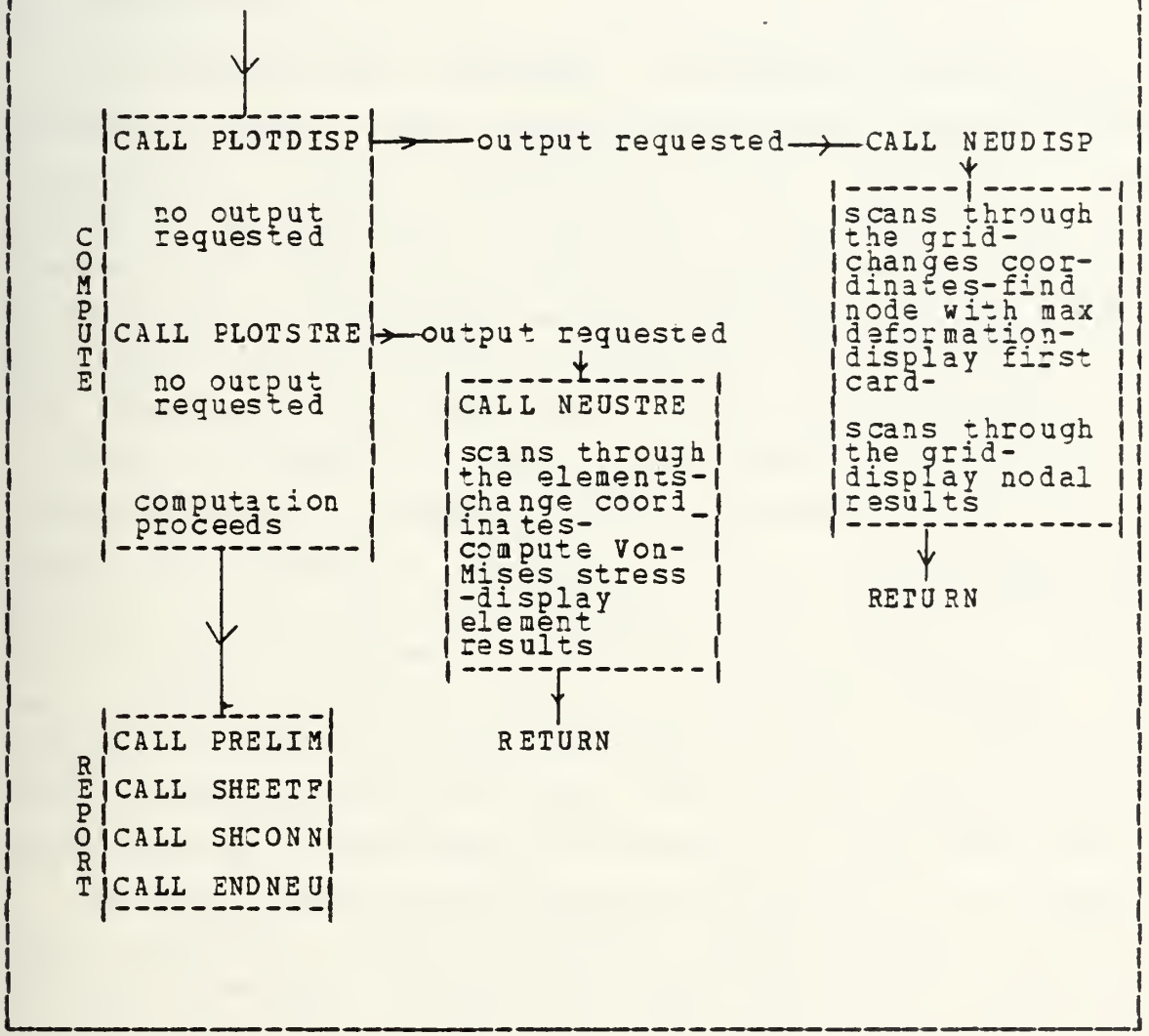
Example:

If five (5) neutral element results files are requested on the input card of a run of 20 steps, five files FOR016.DAT;1 to FOR016.DAT;5 will be created, corresponding to time steps 4 to 20 respectively.

For the displaying of element and nodal points results, the user will refer to [Ref. 7] section 20. The title of the run (first card of EPSA input deck) will be displayed on the screen along with the time at which the results have been requested.

TABLE IV

Structure of the Translator



IV. DESCRIPTION OF THE UNDERWATER EXPLOSION

A. PRESENTATION

An explosion is a chemical or nuclear reaction in a substance (water) that converts an original material into other products plus a significant amount of energy. The process of the explosion produces very high temperatures and pressures and occurs with extreme rapidity. As the result of the explosion, the initial mass of explosive becomes a very hot mass of gas at tremendous pressures; it is then obvious that these conditions will affect the surrounding medium.

The fact that the water is compressible leads to the conclusion that the pressure applied at some location in the liquid will propagate through it as a wave disturbance [Ref. 9]. It must be pointed out that the pressures involved in an underwater explosion are usually so large that the wave velocity cannot be assumed independent of pressure. Thus, the small amplitude wave theory detailed in [Ref. 9] does not apply and this will be the source of many complications in describing the behavior of the shock wave.

The first cause of disturbance to the water in an underwater explosion is the occurrence of the pressure step at the water boundary. Immediately upon its arrival, the pressure begins to be relieved by an intense pressure wave and outward motion of the water. For explosives like TNT or for nuclear explosions, the pressure rise can be considered as a discontinuous step, and is then followed by a roughly exponential decay. The duration of the phenomenon is of the order of a few milliseconds (more for nuclear explosions). Once initiated, the pressure disturbance is propagated radially outward as a compression wave, also called a shock wave because of the steep pressure step at its front.

Close to the explosion, the velocity of the wave is several times the "acoustic" speed of the small amplitude theory (c=5000 ft/sec); but approaches this limit very rapidly as the wave advances outward.
The pressure level in the wave falls more rapidly with the radial distance than what is predicted with the small amplitude theory, but approaches this behavior at large distances.

B. BUBBLE EFFECT

As a result of the explosion, the initial mass of explosives becomes a hot mass of gas at tremendous pressures. After the principal part of the shock wave has been emitted, the gas pressure is considerably decreased but is still much higher than the equilibrium pressure. The water in the immediate region of the sphere or "bubble" of gas has a large outward velocity and the diameter of the bubble increases rapidly. The expansion continues and the internal gas pressure decreases gradually, but the motion persists because of the inertia of the outward flowing water. When the gas pressure falls below the equilibrium value, the pressure defect brings the outward flow to a stop and the boundary of the bubble begins to contract at an increasing rate. The inward motion continues until the compressibility of the gas reverses the motion. Thus, the inertia of the water together with the elastic properties of the gas provide the necessary conditions for an oscillatory system. The oscillations of the gas sphere may persist a number of cycles, ten or more oscillations having been detected in favorable cases.

C. SURFACE EFFECTS

In the case of explosions occurring at shallow depths, surface effects will complicate the aforementioned sequence of events. When the shock wave hits the surface, the atmosphere cannot supply appreciable resistance by compression. As a result, a reflected wave with a negative pressure satisfying the zero-pressure condition at the surface is formed (figure 4.1). Thus, the resultant pressure observed is the sum of the direct and reflected pressures. Therefore, the reflected wave arriving at point A will create a sudden drop of the pressure to a smaller value. This is known as the "cut-off" phenomenon, typical of free surface effects (figure 4.2).

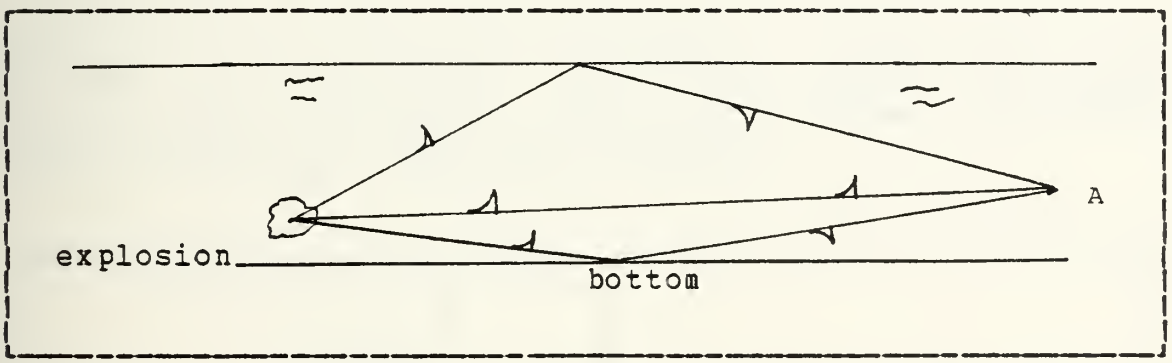


Figure 4.1 Surface Effect on a Shock Wave.

D. PRESSURE DETERMINATION

As detailed in a previous section, the pressure decay at any point is roughly exponential so that it can be written:

$$P_t(A) = P_0(A)\, e^{-t/\theta} \qquad (4.1)$$

It has been found that the fundamental descriptors of an underwater explosion attack are the charge size (equivalent weight of TNT) and the charge standoff (shortest distance between charge and target). Theoretical developments about the spherical wave detailed in [Ref. 10] have shown that the peak pressure P_m at any point can be reasonably approximated by:

$$P_m = K_1 (W^{1/3}/R)^{A1} \qquad (4.2)$$

where W is the charge size in pounds of TNT and R is the standoff distance in feet.
It has been shown as well:

$$\theta = K_2 W^{1/3} (W^{1/3}/R)^{A2} \qquad (4.3)$$

K_1, K_2, A_1, A_2 are empirically determined factors that depend on the type of explosives used. Their values for several types of explosives are given on table V.

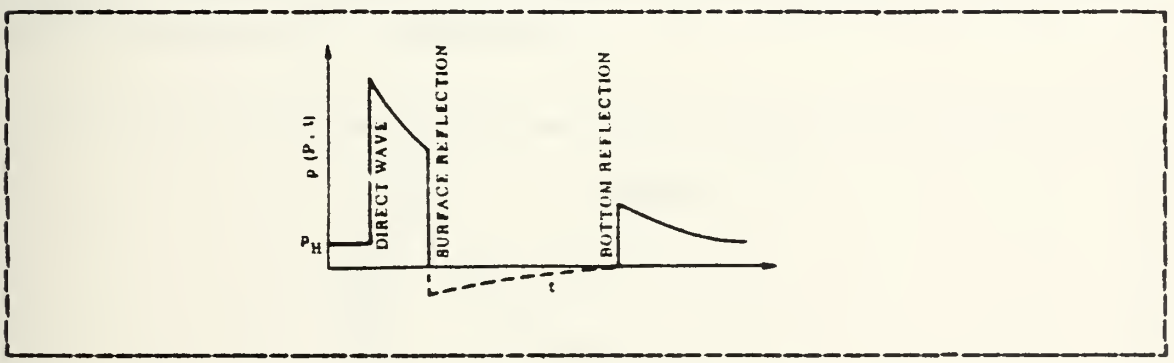


Figure 4.2 Cut-Off Phenomenon.

TABLE V

Explosion Parameters

		HBX-1	TNT	PENT	NUKE
Pmax	K_1	22347.6	22505	24589	$4.38X10^6$
	A_1	1.144	1.18	1.194	1.13
Decay Constant	K_2	.056	.058	.052	2.274
	A_2	- .247	- .185	- .257	- .22

E. THE EXPLOSION IN EPSA

EPSA features two different way of describing underwater explosions:

-A discrete form in which the user inputs the pressure history at a finite number of times.

-A functional form that uses equations 4.2 and 4.3 .The program requests then the various coefficients and parameters describing the explosion.

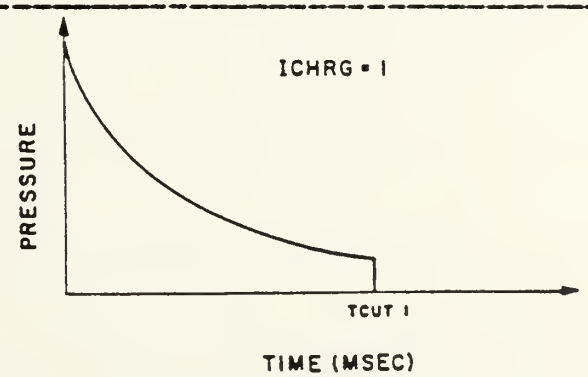


Figure 4.3 Incident Pressure Decay.

The presence of free-surface effects can also be accounted for with the input of a cut-off time after which the incident pressure is set to zero (figure 4.3).

V. ANALYSIS AND RESULTS

A. MODELS STUDIED

In order to compare the results of the numerical analysis with the experimental data, all attention was focused on the Explosive Power Meter (EPM) model for which field tests had been conducted. The EPM model is a stiffened cylinder with end caps whose dimensions are given in figure 5.1 .

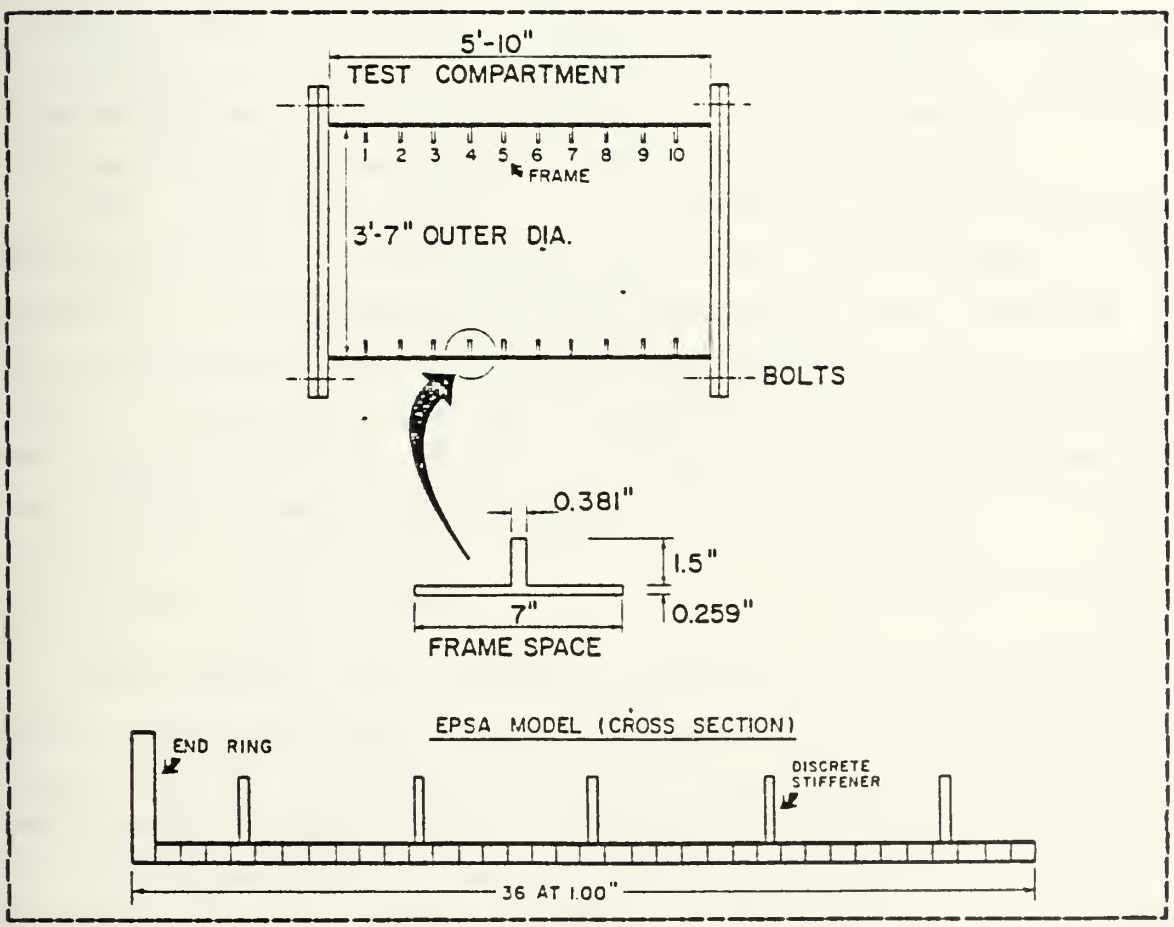


Figure 5.1 Explosive Power Meter.

Rather than modelling the end caps with additional sheets , it was found to be more efficient to take into account the behavior of the end caps using two rigid end blocks (figure 5.1). The effects of the explosion on the deformation of the end blocks is negligible compared to the deformation of points located outside of the end blocks. Therefore, it was assumed that the motions of the end blocks were pure rigid body displacements.

In order to gain some insight into the influence exerted by the stiffeners and the end blocks, a preliminary analysis was conducted on a ring stiffened cylinder without end blocks as well as on an unstiffened cylinder .

In addition to the study of the EPM model, the influence that the location of the stiffeners had on the deformations throughout the shell was evaluated. By performing a comparative analysis of the deformations, it was intended to minimize and control the damage caused to the structure.

The cylinders tested were subjected to an explosion occurring at the distance R= 200" from the cylinder. The location of the explosion was symmetrical with respect to the longitudinal and transverse axis of the cylinder (figure 5.2). A spherical type TNT explosion of intensity W=50 lb was selected for the study. It was therefore determined by the following parameters (chapter IV) :

$A_1 = 1.18 \quad A_2 = -.185$

$K_1 = 22505 \quad K_2 = .058$

The symmetry with respect to the xy and yz plane has been taken advantage of by modelling one quarter of the model. After a certain amount of sensitivity analysis was performed on simple grids, a finite element grid consisting of 1517 nodes and 1440 elements was selected (figure 5.3) . For each of the cylinders studied, the time step chosen was equal to $\Delta t = 3.10^{-6}$ s . The explosion process was studied

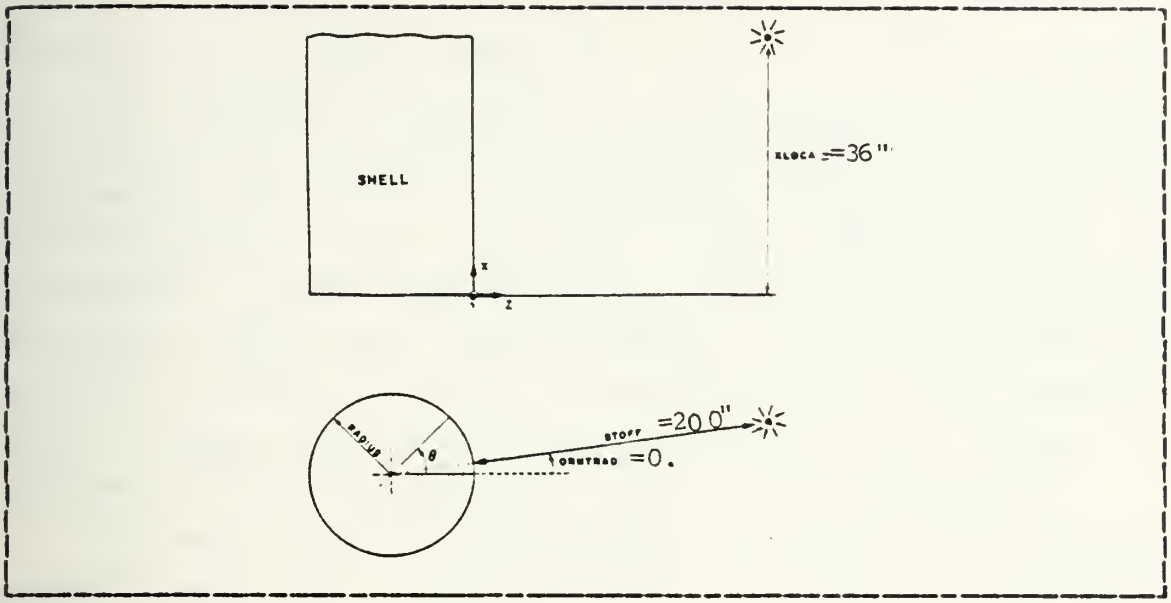


Figure 5.2 Explosion Location.

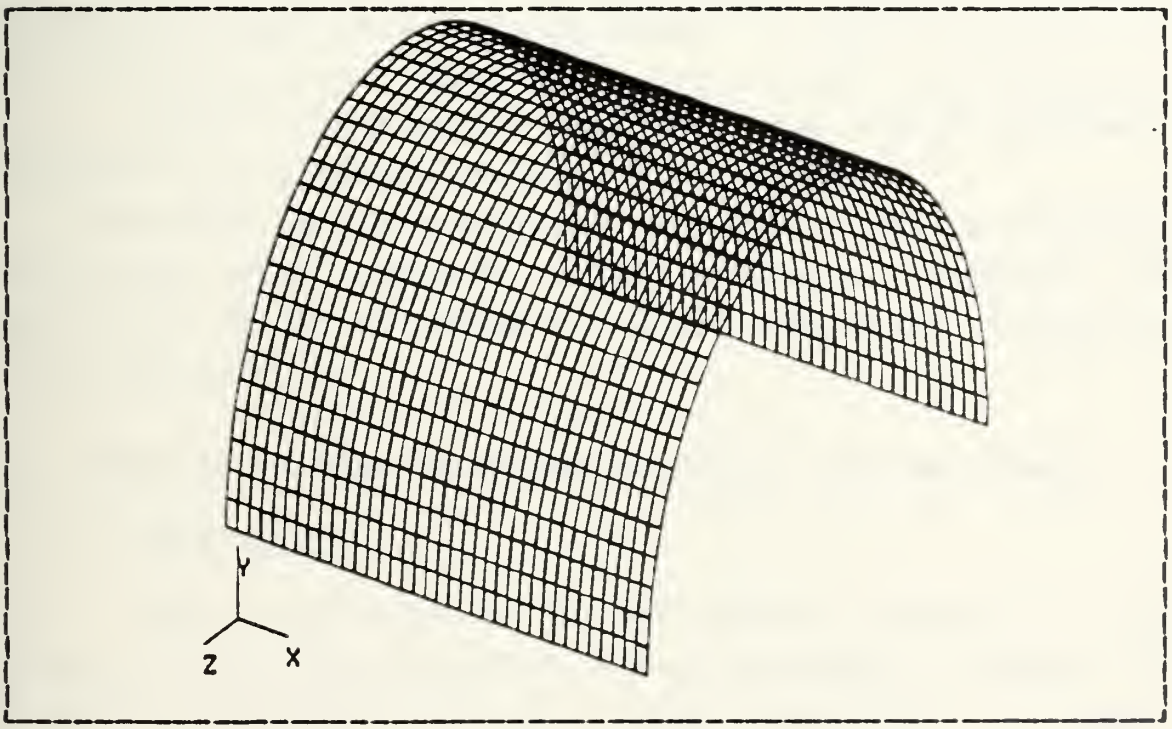


Figure 5.3 FEM Discretization 1517 nodes, 1440 elements.

over a time interval of 800 time steps, that allowed for a shock and after-shock analysis.

The displacements computed by EPSA consist of a combination of rigid body displacements and pure deformations. The deformation modes are of significant interest since they may induce buckling and even lead to the destruction of the structure. As mentioned earlier, the very stiff end blocks have pure rigid body displacements. The displacement of each node of the end block was subtracted from the displacement of each node of the corresponding row, giving the component of the displacement due to pure deformation.

For each of the aforementioned cylinders, the deformed geometry and the color-filled contour plots of von Mises stresses as well as normal displacement were displayed. Using identical color assignments, the progressive gross evolution of the previous quantities were evaluated so as to allow for a comparative analysis of the evolution of physical parameters throughout the shell.
Color-filled contour plots allow for a global representation of a quantity distribution and have been found extremely valuable in the interpretation of the results.

For printing and processing reasons, it was not possible to include color pictures in this document. Instead, the contour plots of the physical quantities under study have been included.

B. ANALYSIS OF RING STIFFENED CYLINDERS WITH END BLOCKS

1. EPM Model

The contour plots of von Mises stresses at time steps 20, 60, 100, 140, 180, 200 are provided on figure A.1 to A.6 . As the shock wave hits the structure, it appears as if the stresses propagate through the shell and reach their maximum value fairly quickly in 50 time steps. After

100 time steps, when the structure is fully enveloped by the shock wave, the region close to the end block becomes heavily stressed (figure A.4). In addition, some concentration of stresses at the locations of the stiffeners can be seen (figure A.5). At later time steps, the pressure becomes decreased and there is a relaxation of stresses. However, the region close to the end block as well as some spots located around the stiffeners remain heavily stressed, which may indicate local buckling (figure A.6).

2. Controlled Deformations

In order to obtain a more uniform distribution of displacements, the stiffeners have been shifted towards the end blocks. The time history evolution of the displacements was studied over an interval of 800 time steps for the EPM model as well as for the model with shifted stiffeners called EPM2. The contour plots of normal displacements at time steps 200, 400, 600, 800 for both models are provided on figure A.7 to A.14

The EPM model shows a significant concentration of deformations occuring, even at late time steps (figure A.10), indicating a possibility of buckling. Although unexpected, the fact that the region close to the end blocks undergoes the most severe deformations has been confirmed by experimental data. A possible explanation to this phenomenon is that the inertia of the cross-section of the cylinder is relatively uniform along the cylinder, except at the end blocks where it jumps to a much higher value. This discontinuity in the inertia results in concentrations of stresses that cause the aforementioned phenomenon.

The cross-section inertia of the EPM2 model increases more smoothly because of the distribution of stiffeners along its axis. Thus, the concentration of stresses has a lower magnitude and the region close to the end block

suffers less damage than in the EPM case. It can also be seen that the deformations are more uniformly distributed along the axis of the cylinder. Above all, the deformations at the late time steps are not as large, indicating that the chances of buckling are significantly lower for the EPM2 model (figure A.14).

Therefore, by performing an optimization of the location of the stiffeners, the designer can counteract the concentrations of stresses and the buckling phenomena that occur in the region close to the end blocks. It is believed that controlled deformations can have a very significant influence on the survivability of a structure submitted to a shock wave.

C. ANALYSIS OF UNSTIFFENED AND RING-STIFFENED CYLINDER

It was decided to study the progressive gross responses of an unstiffened cylinder as well as that of a ring-stiffened cylinder without end-blocks. Both cylinders have the same external dimensions as the EPM model. The ring-stiffened cylinder is similar to the EPM model except for the fact that the end-block has been replaced by a standard stiffener. The evolution of von Mises stresses at time steps 40, 80, 100 is provided in figures A.15 through A.17 for the unstiffened cylinder. The evolution of von Mises stresses at time steps 40, 80, 100, 150 is provided in figures A.18 through A.21 for the ring-stiffened cylinder without end-blocks.

For the unstiffened cylinder it is observed that the stresses propagate quickly throughout the shell and that within a hundred time steps an instability phenomenon occurs showing the existence of local buckling (figure A.17). At later time steps the buckling spreads over the entire stiffened shell.

The evolution of von Mises stresses in figures A.18 through A.21 shows that the ring-stiffened cylinder can withstand much higher stress levels than the unstiffened shell. This result was expected since the stiffeners provide the stiffness required to withstand higher loads. At time step 100 the unstiffened shell is already subjected to local instability characterized by a "hard spot" in the middle of the model (figure A.17). On the other hand, the stresses in the stiffened cylinder are much more evenly distributed throughout the model, with high amounts of stresses concentrated around the locations of the stiffeners (figure A.21).

It can also be observed that a significant concentration of stresses occurs at the extremities of the stiffened shell (figure A.21). Recalling that the end-block has been replaced by a standard stiffener, the cross-section of the shell has a greater inertia at the extremities due to the fact that the two stiffeners located at the extremities are close to each other. Therefore this phenomenon is similar to the one encountered when studying the EPM model. However the concentration of stresses for the ring-stiffened cylindrical shell is less significant than for the EPM model, due to a smaller discontinuity in cross-section inertia.

VI. CONCLUSION

A FORTRAN module that merges the finite element code EPSA with the color graphics system PATRAN-G has been designed and succesfully completed. The non-linear elasto-plastic responses of various types of submerged cylindrical shells have been evaluated using the EPSA/PATRAN-G system.

The analysis of the progressive gross responses of a ring-stiffened cylindrical shell with end-blocks is believed to provide useful information regarding the behavior of a submerged structure subjected to an underwater explosion. The influence of the location of the stiffeners on the deformations has been studied and is also believed to be of significant help in the determination of an optimal design that will minimize the damage due to an underwater explosion.

The utilization of the color graphics system in the interpretation of the results of analysis has been found to be an extremely valuable tool. It is the author's belief that the use of color graphics systems will become increasingly important in the analysis of complex phenomena such as underwater explosions on submerged structures.

APPENDIX A
FIGURES

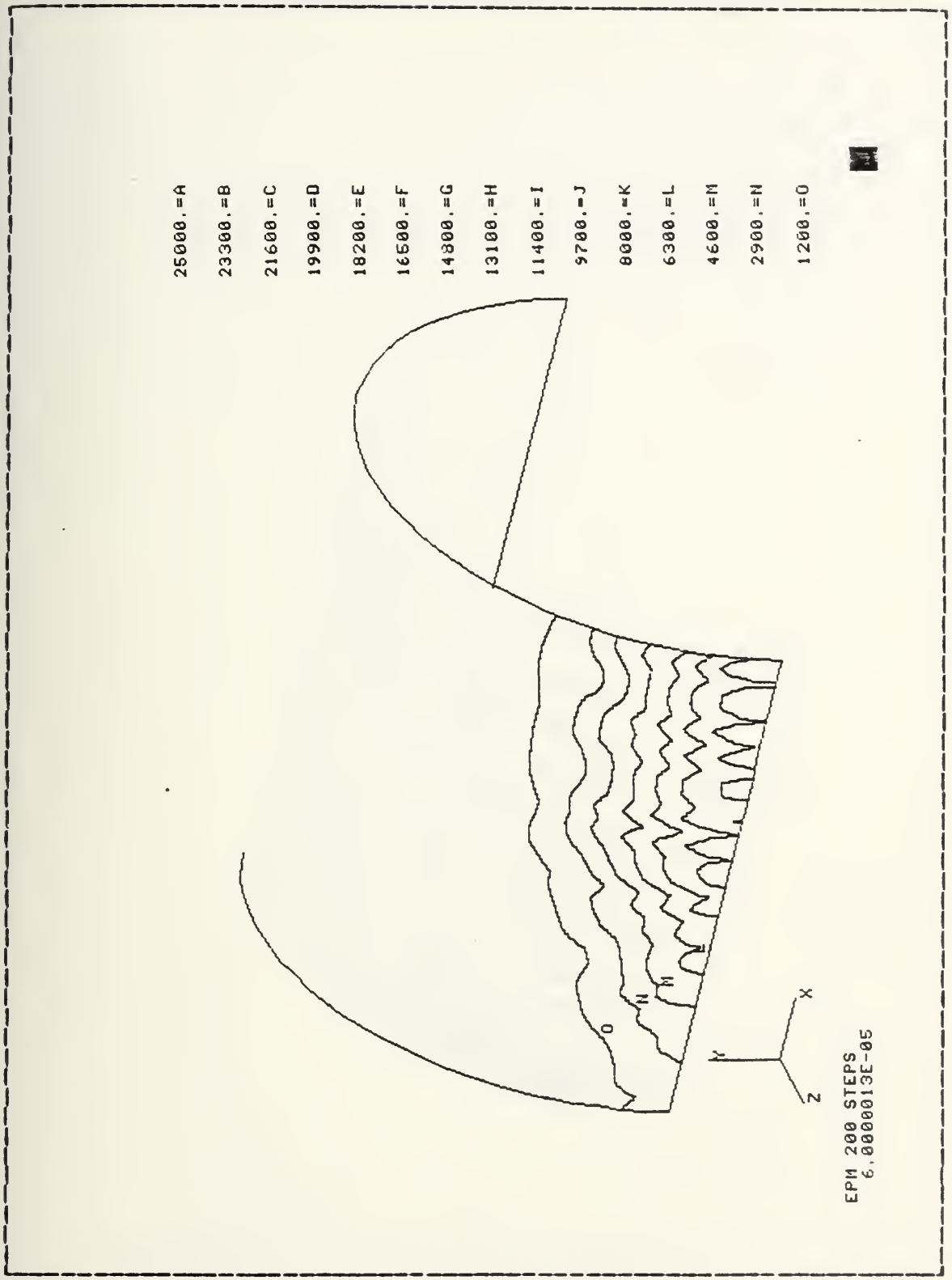


Figure A.1 Von Mises stresses, time step 20 , EPM model.

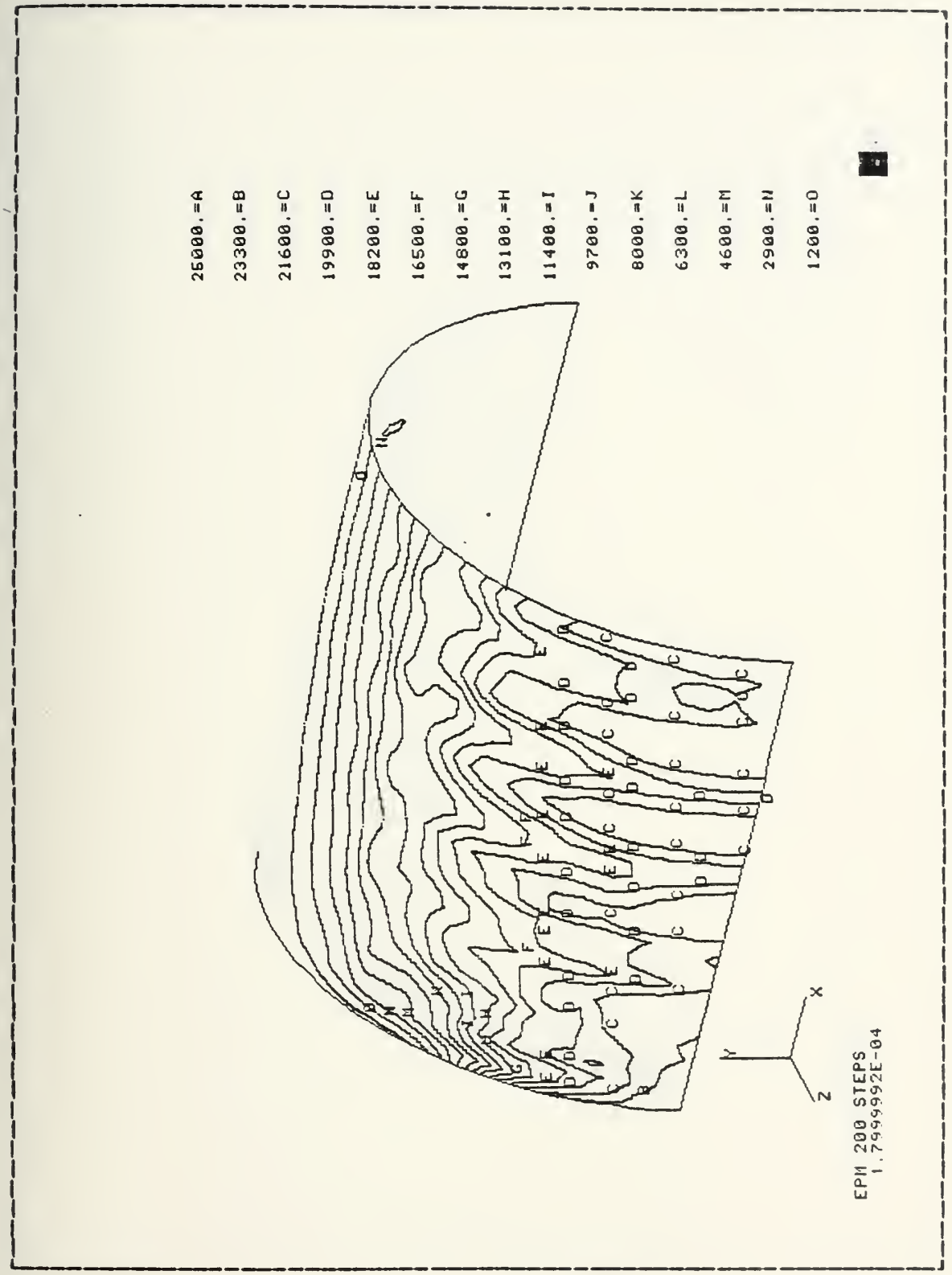


Figure A.2 Von Mises stresses, time step 60 , EPM model.

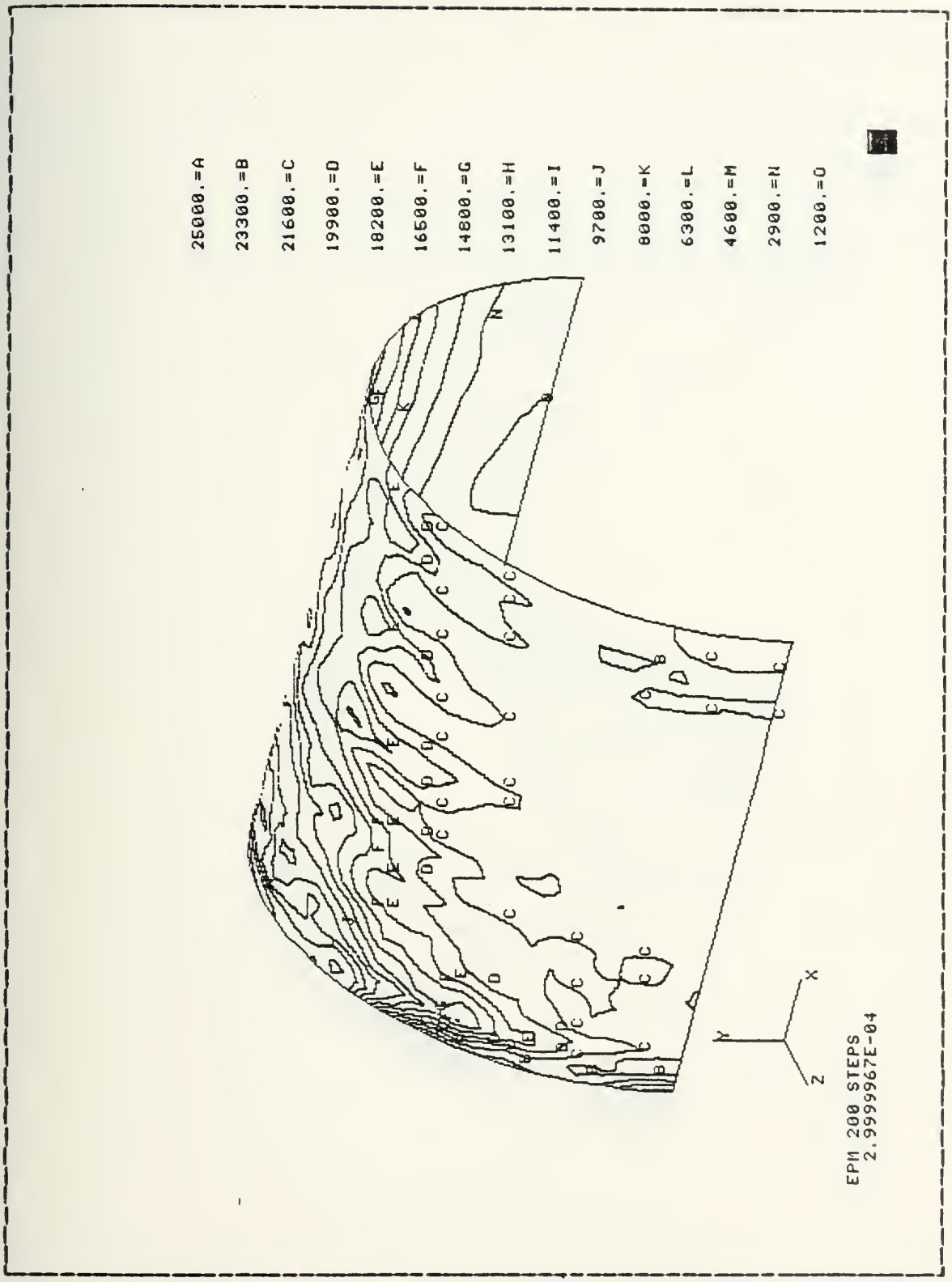


Figure A.3 Von Mises stresses, time step 100, EPM model.

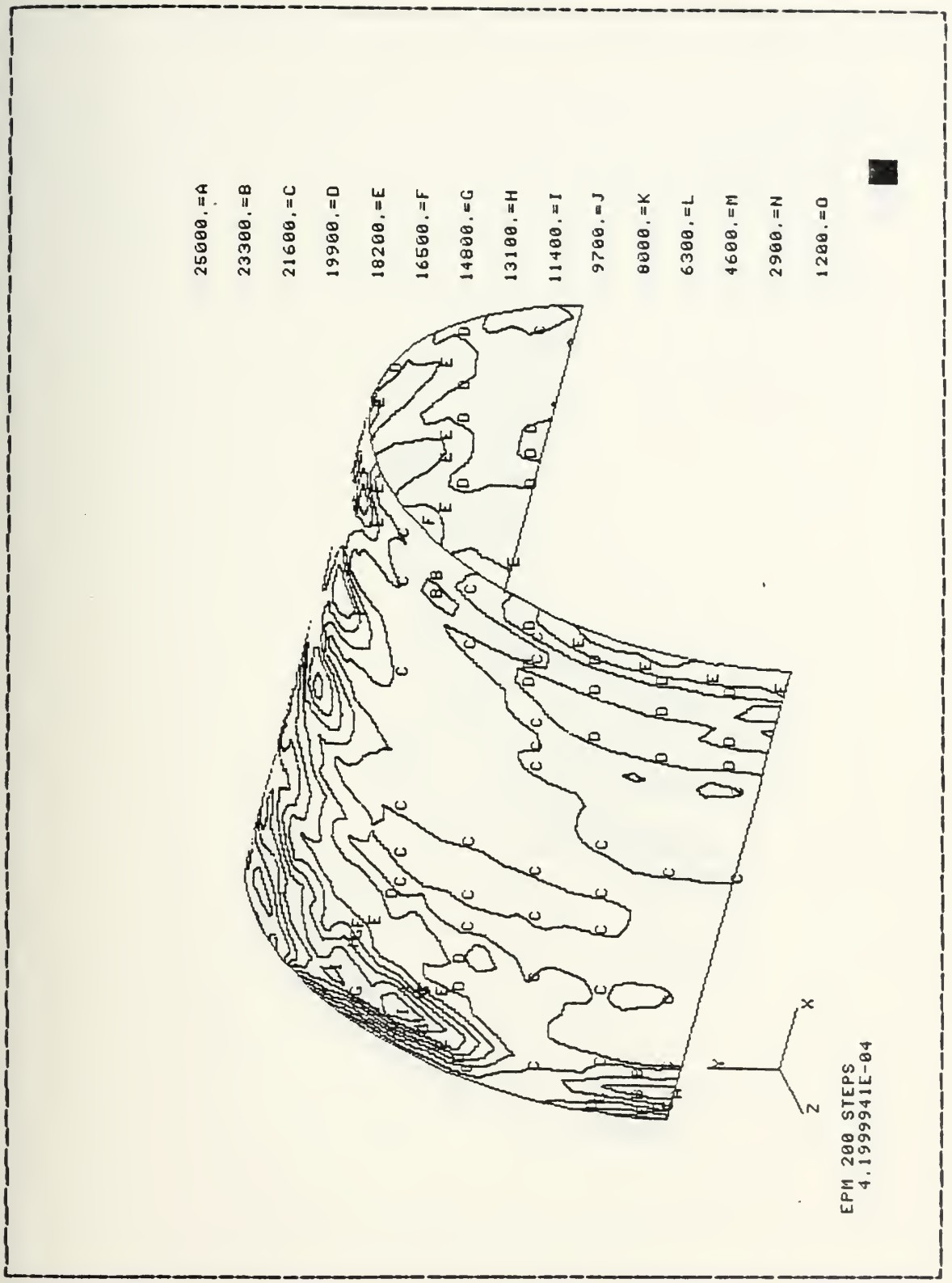


Figure A.4 Von Mises stresses, time step 140, EPM model.

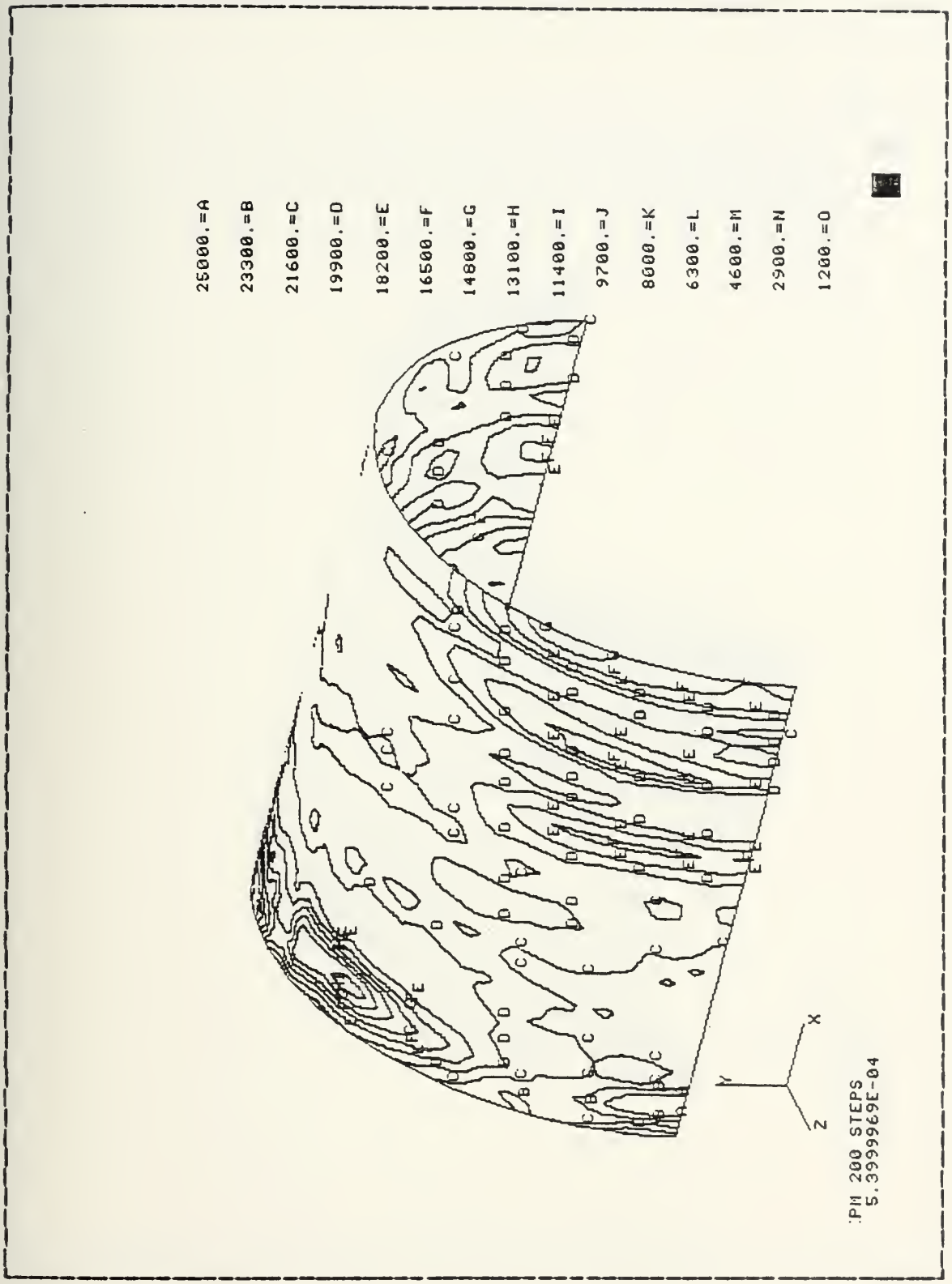


Figure A.5 Von Mises stresses, time step 180, EPM model.

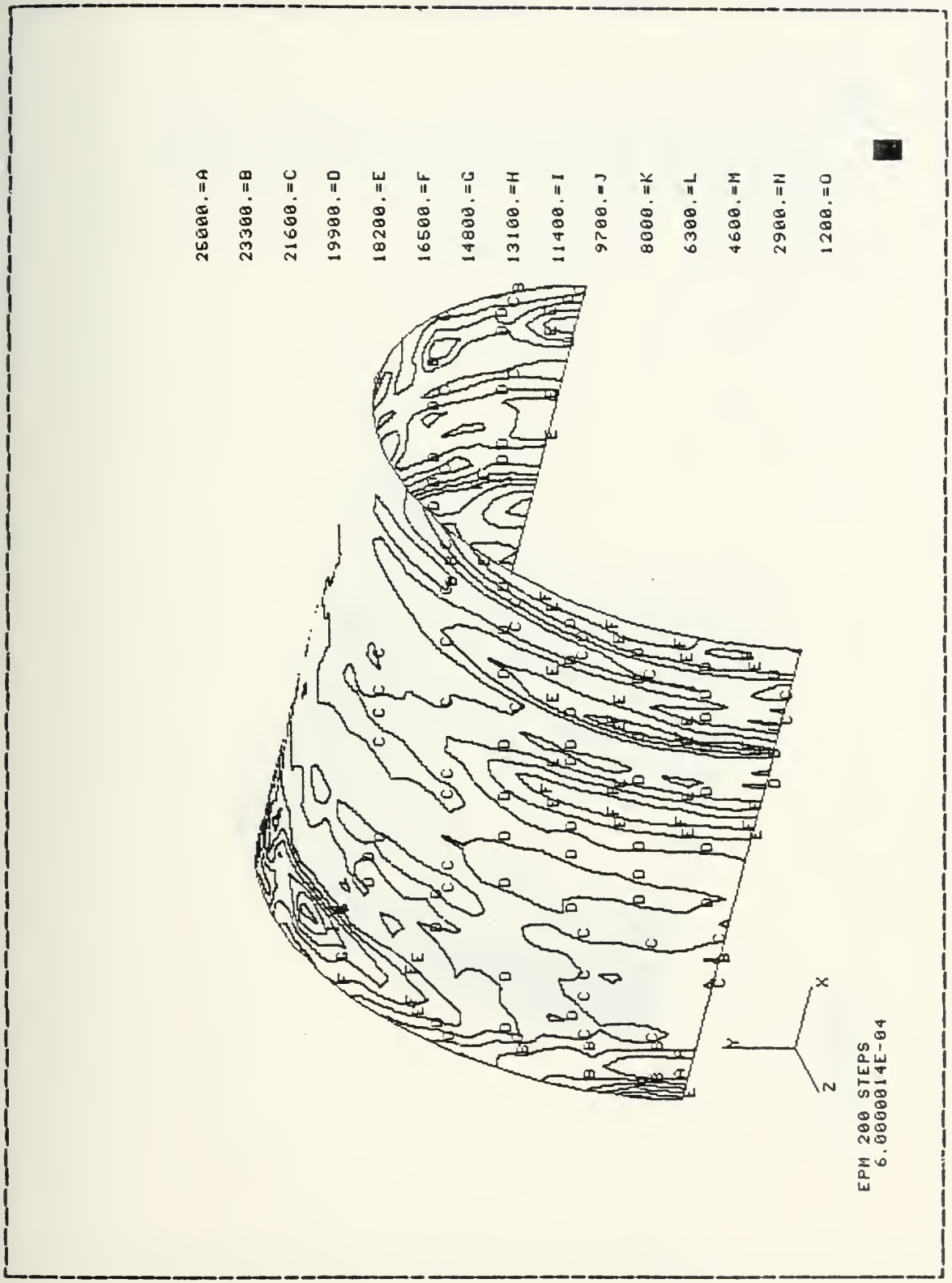


Figure A.6 Von Mises stresses, time step 200, EPM model.

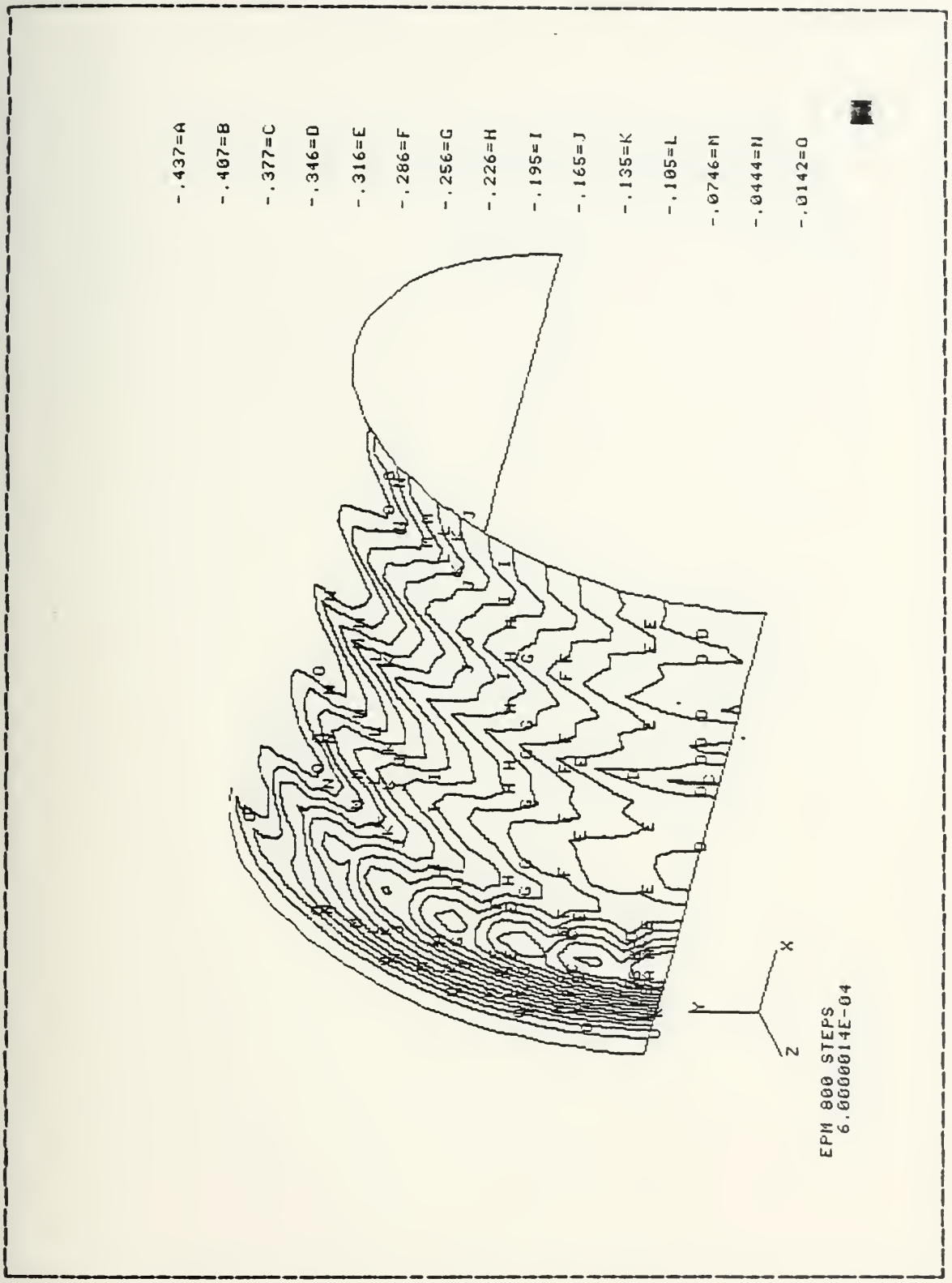


Figure A.7 Normal Displacements, step 200, EPM model.

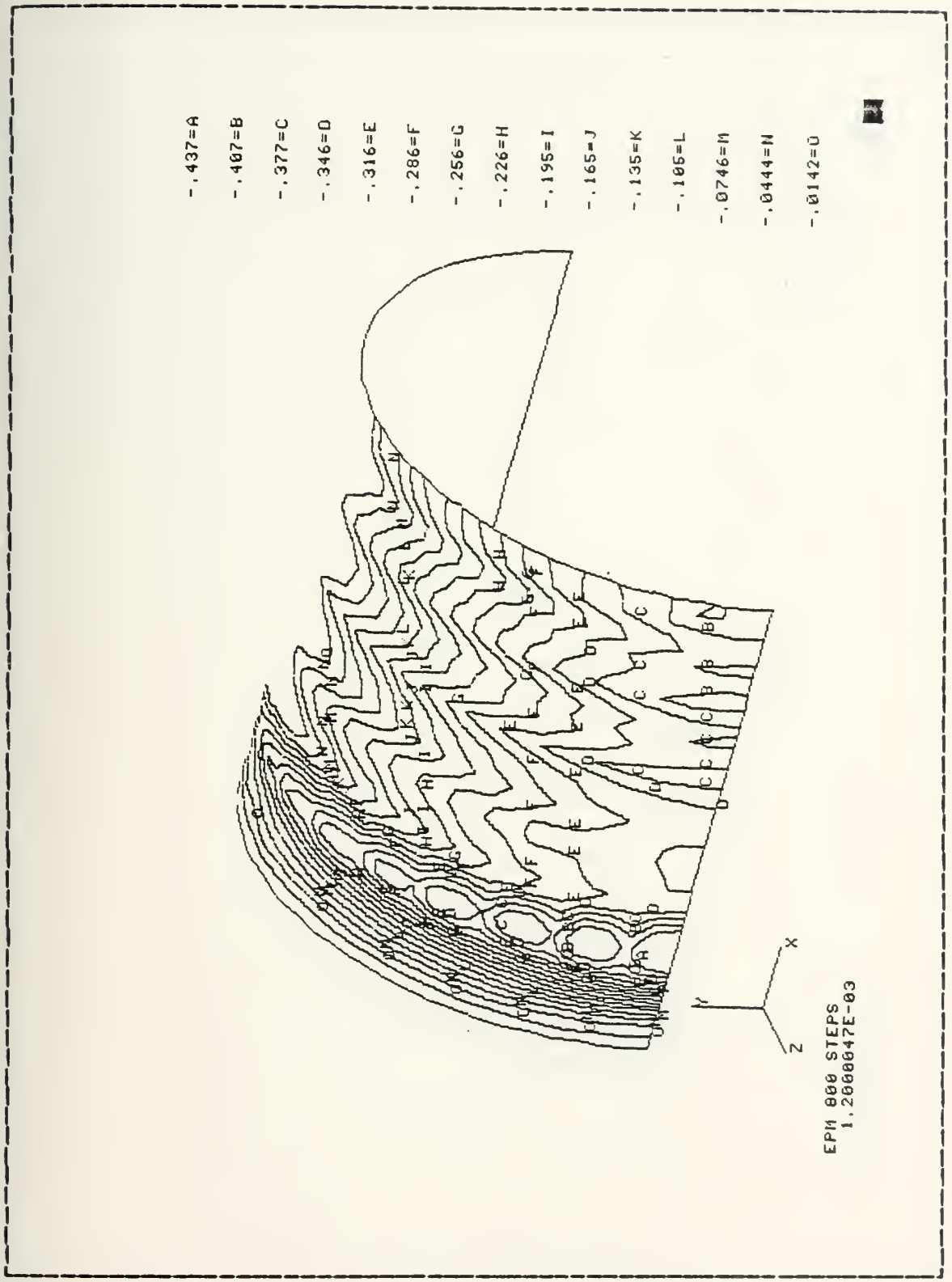


Figure A.8 Normal Displacements, step 400, EPM model.

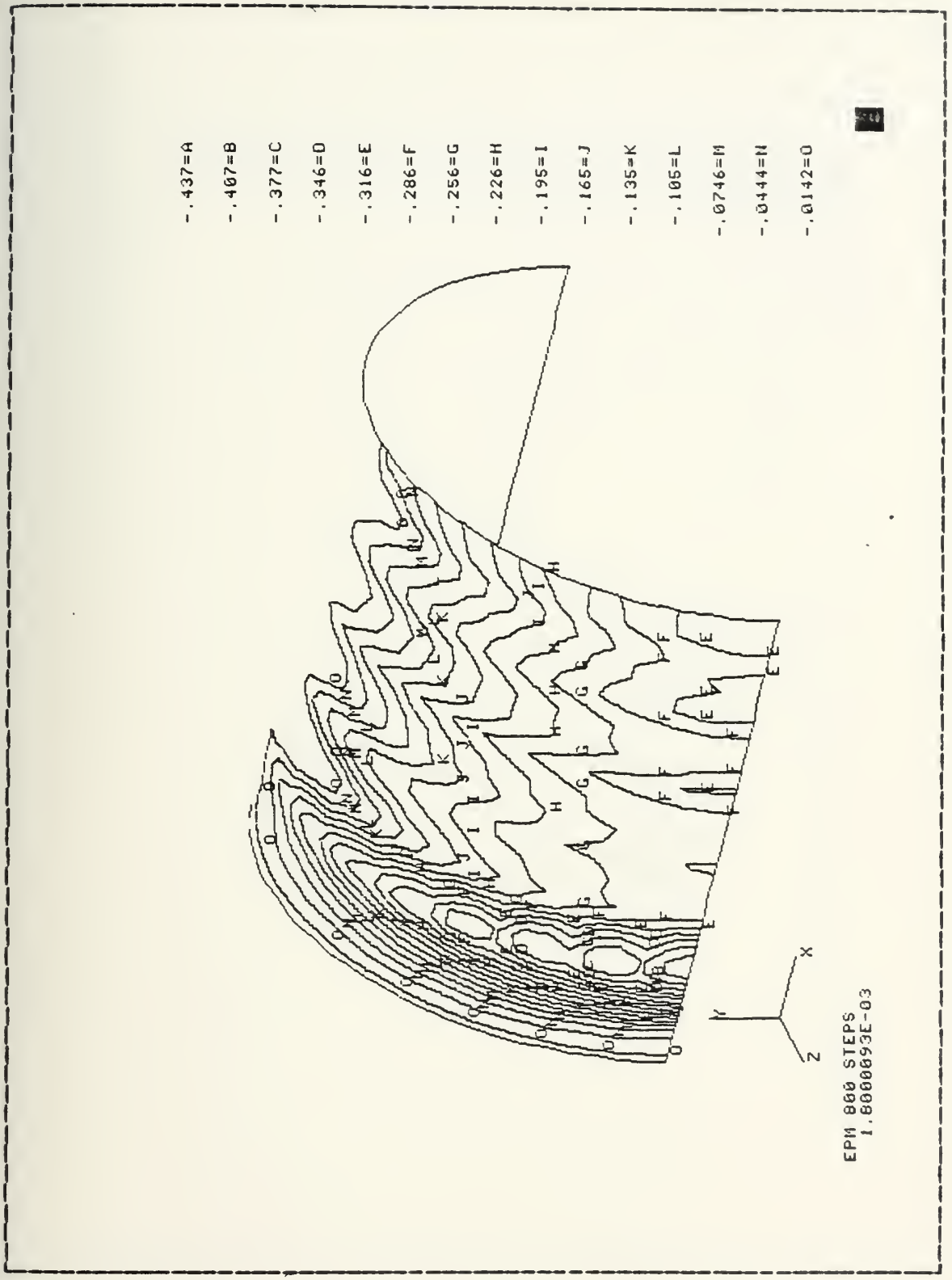


Figure A.9 Normal Displacements, step 600, EPM model.

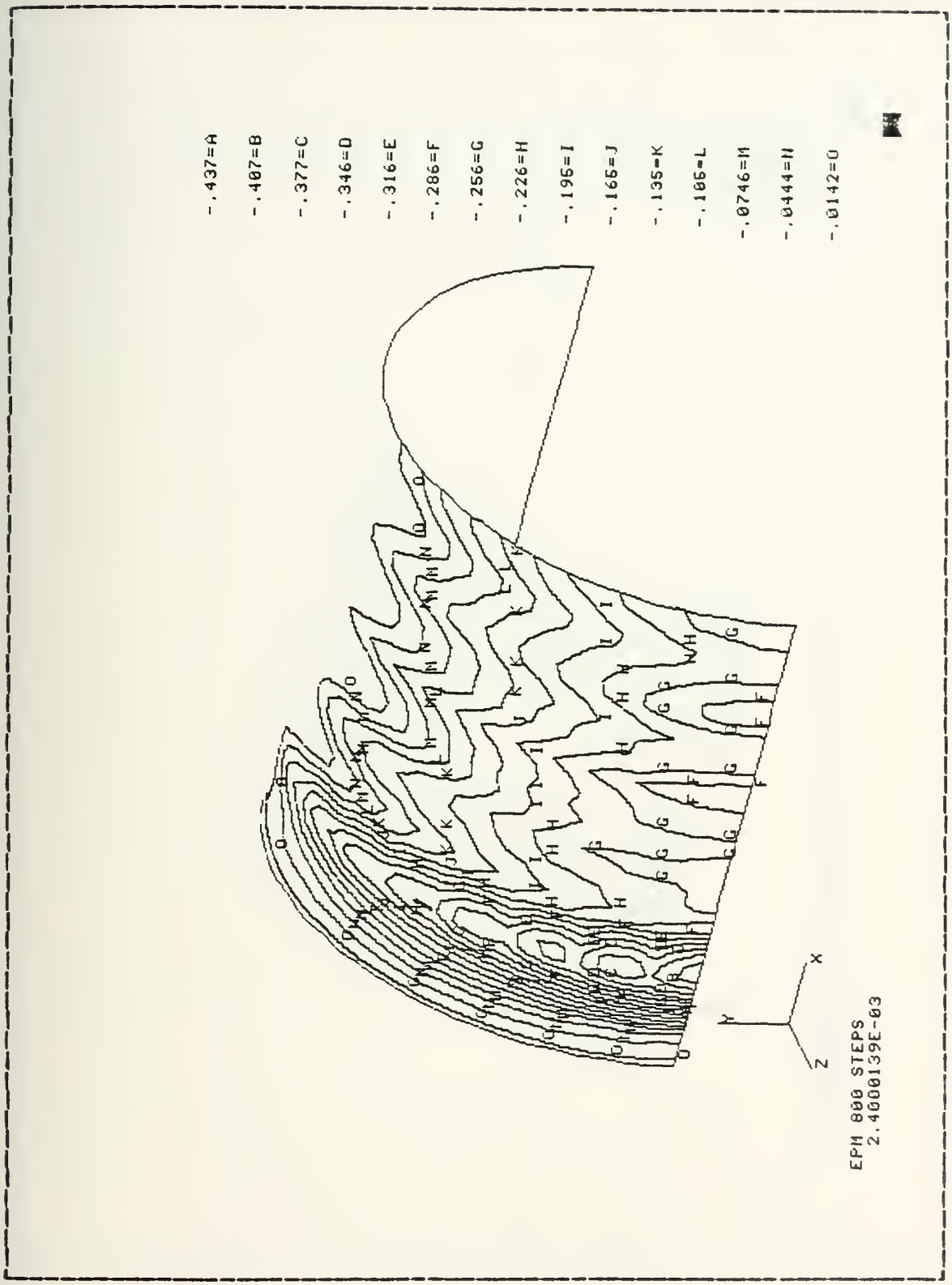


Figure A.10 Normal Displacements, step 800, EPM model.

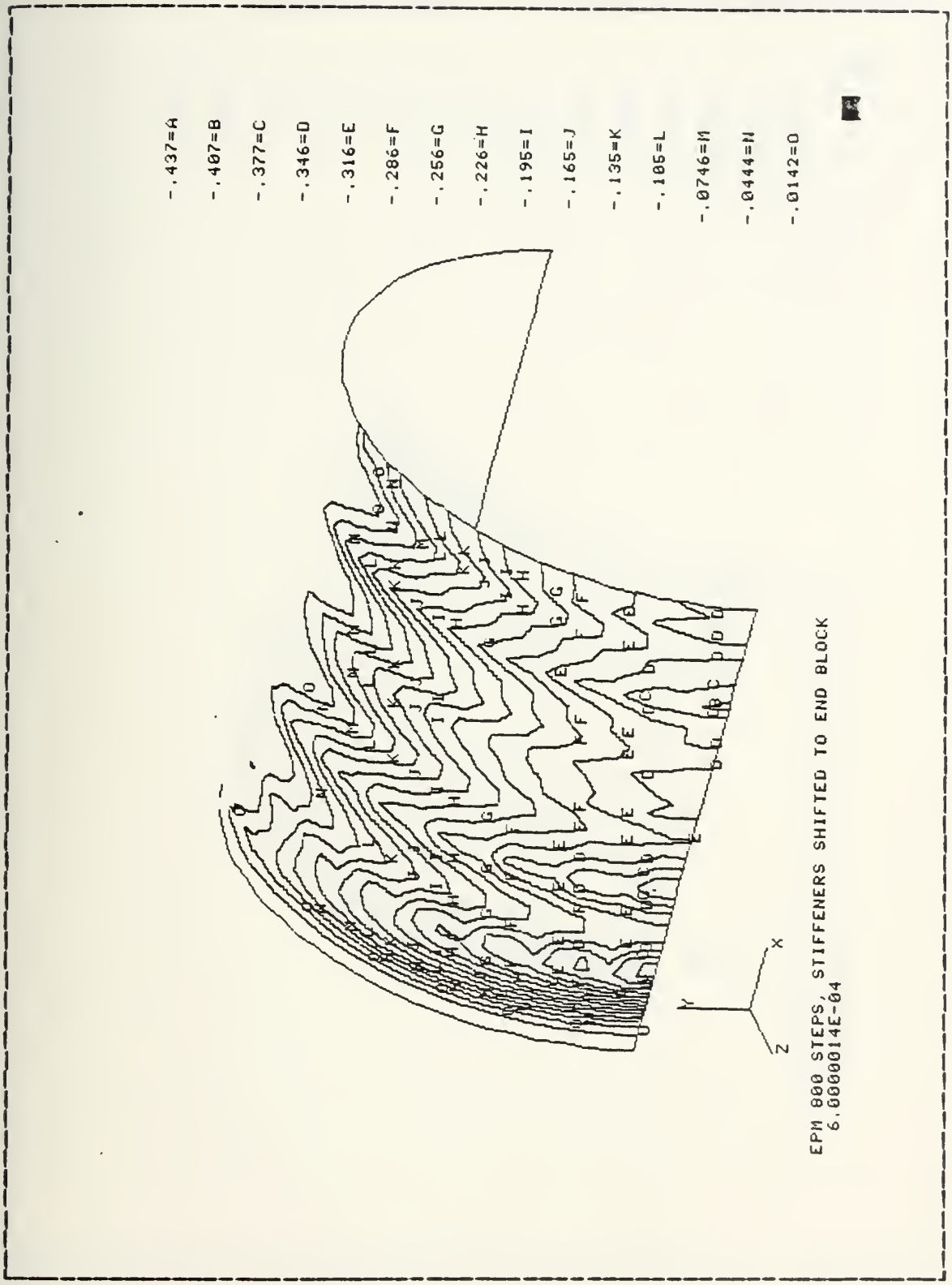


Figure A.11 Normal Displacements, step 200, EPM2 model.

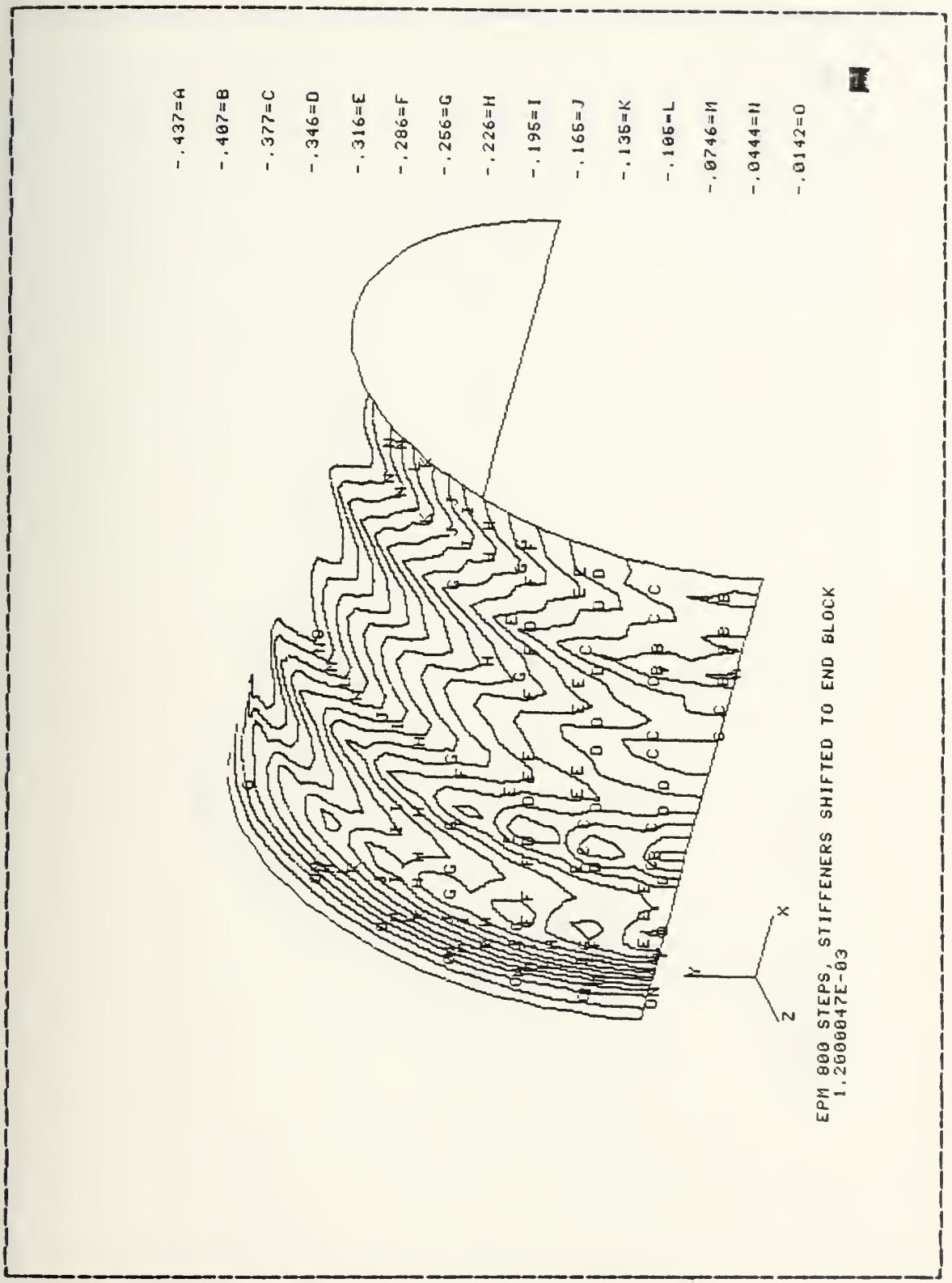


Figure A.12 Normal Displacements, step 400, EPM2 model.

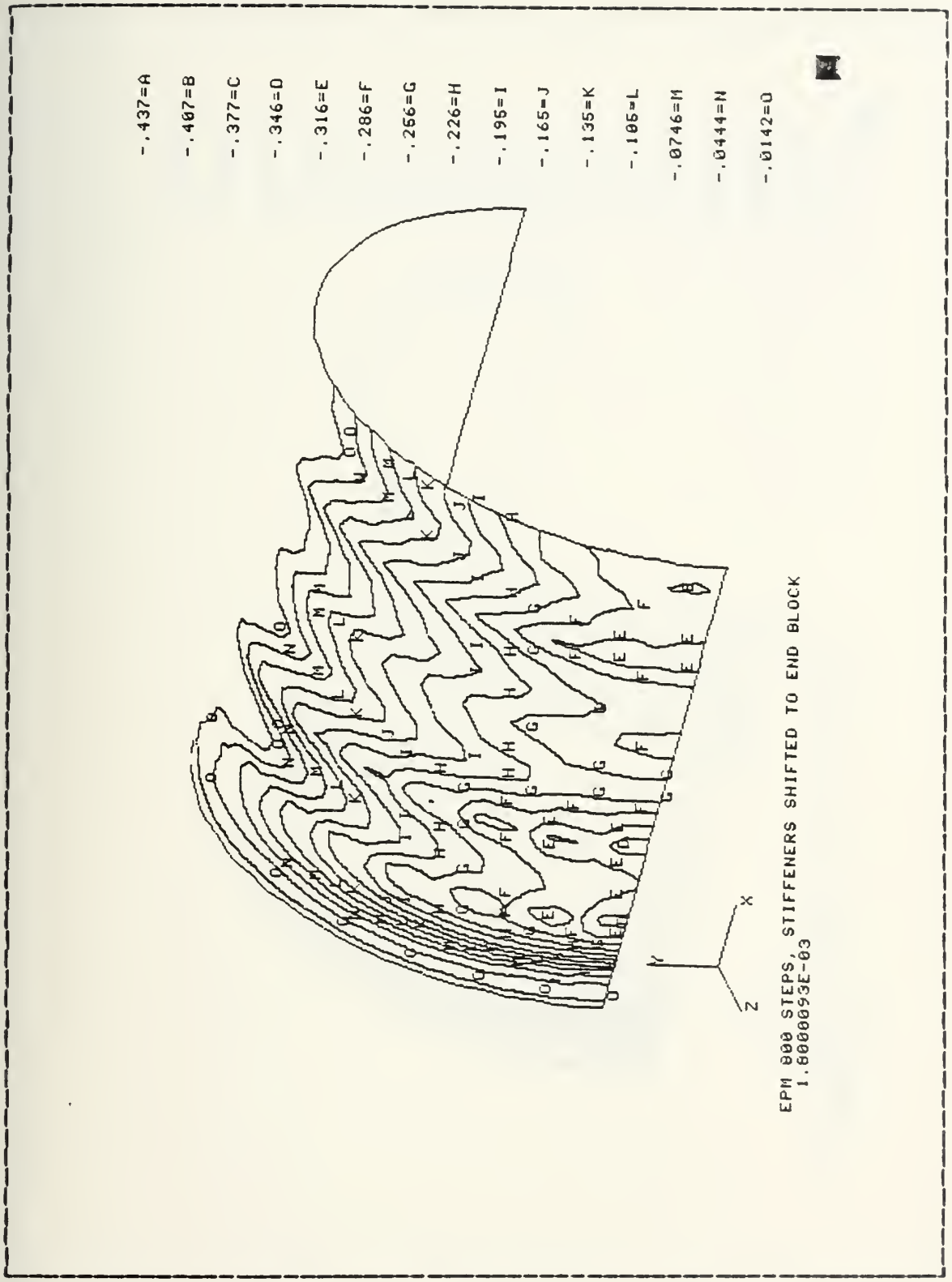


Figure A.13 Normal Displacements, step 600, EPM2 model.

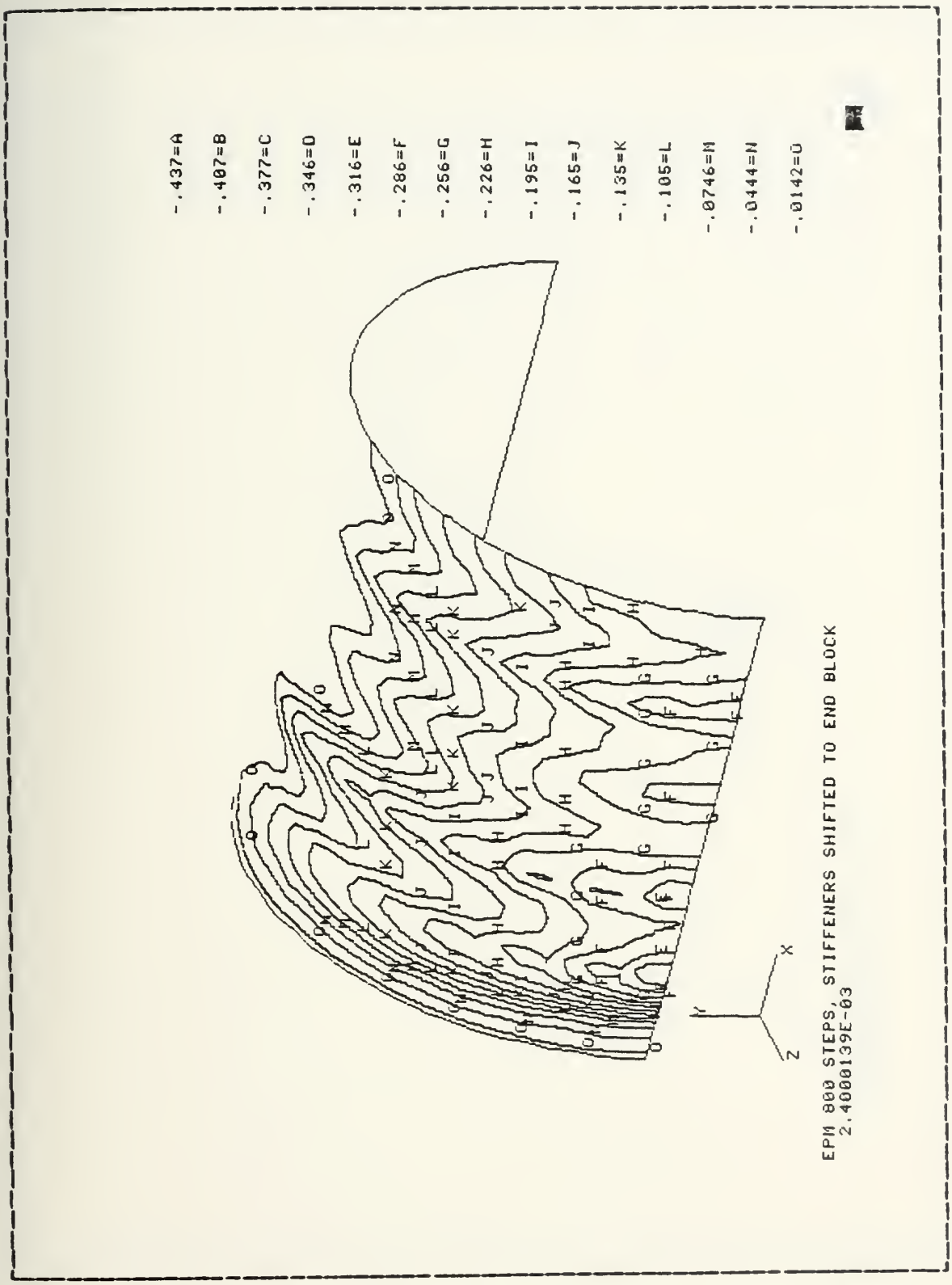


Figure A.14 Normal Displacements, step 800, EPM2 model.

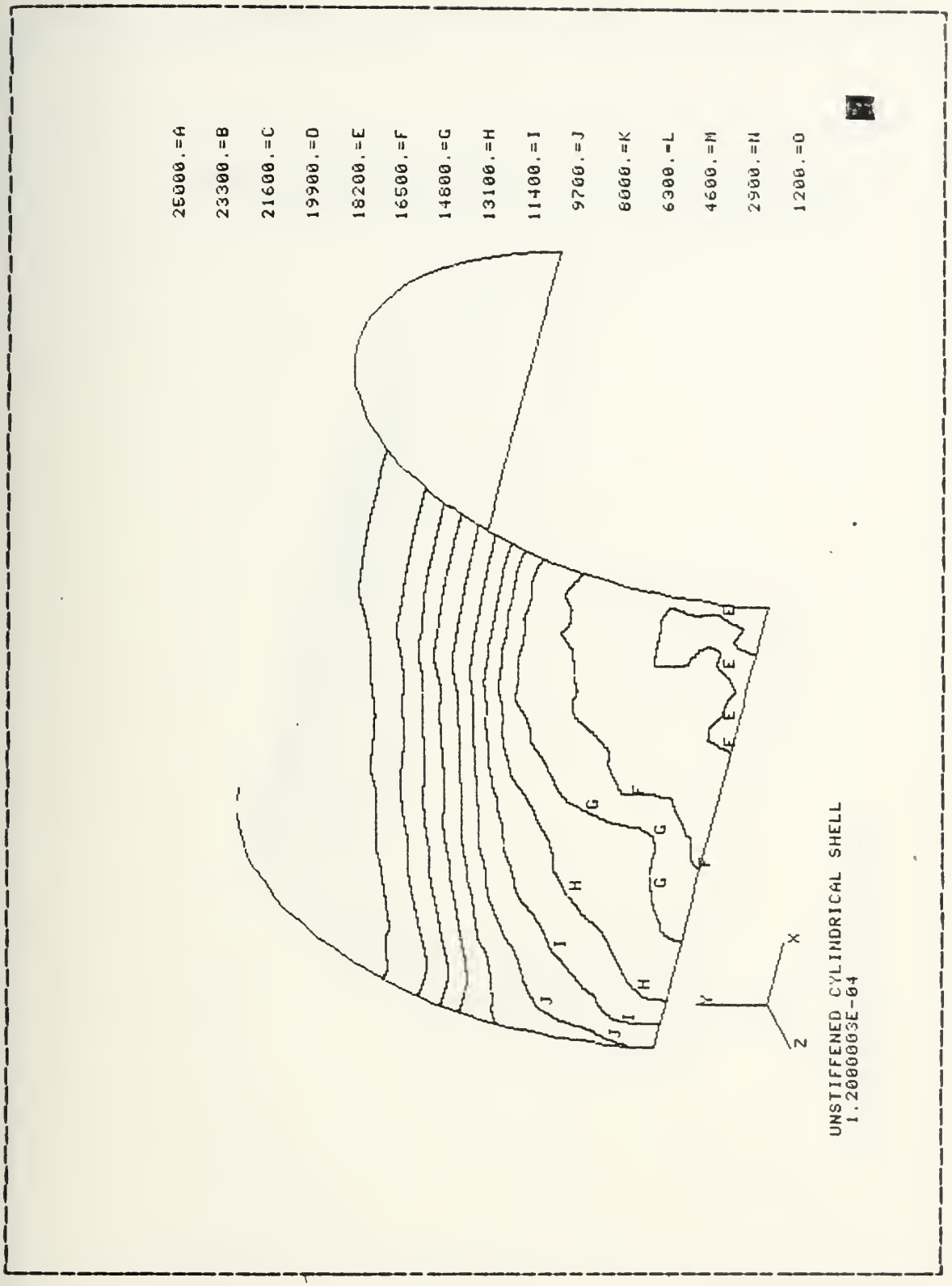


Figure A.15 Von Mises stresses, step 40 , unstif. shell.

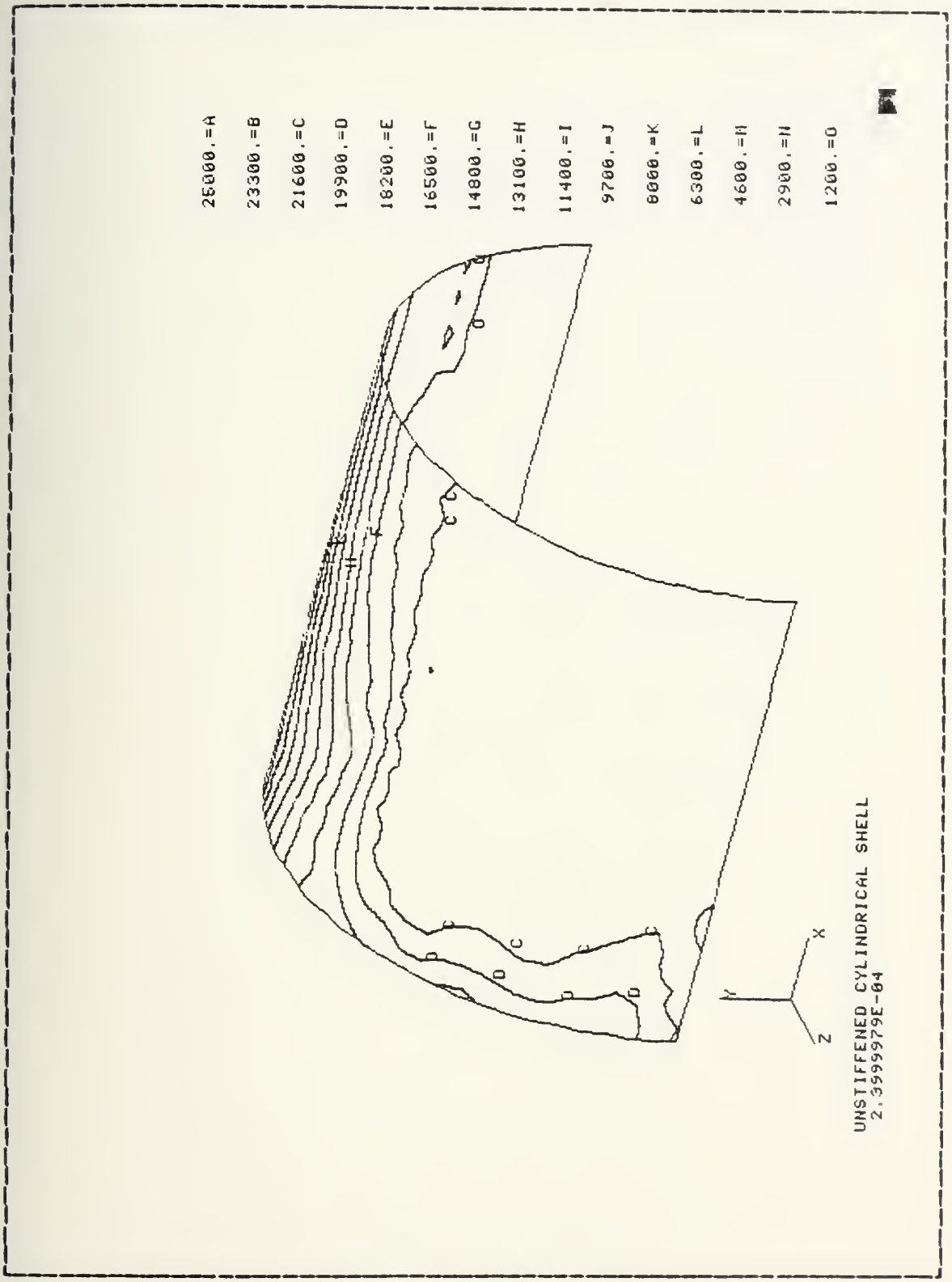


Figure A.16 Von Mises stresses, step 80 , unstif. shell.

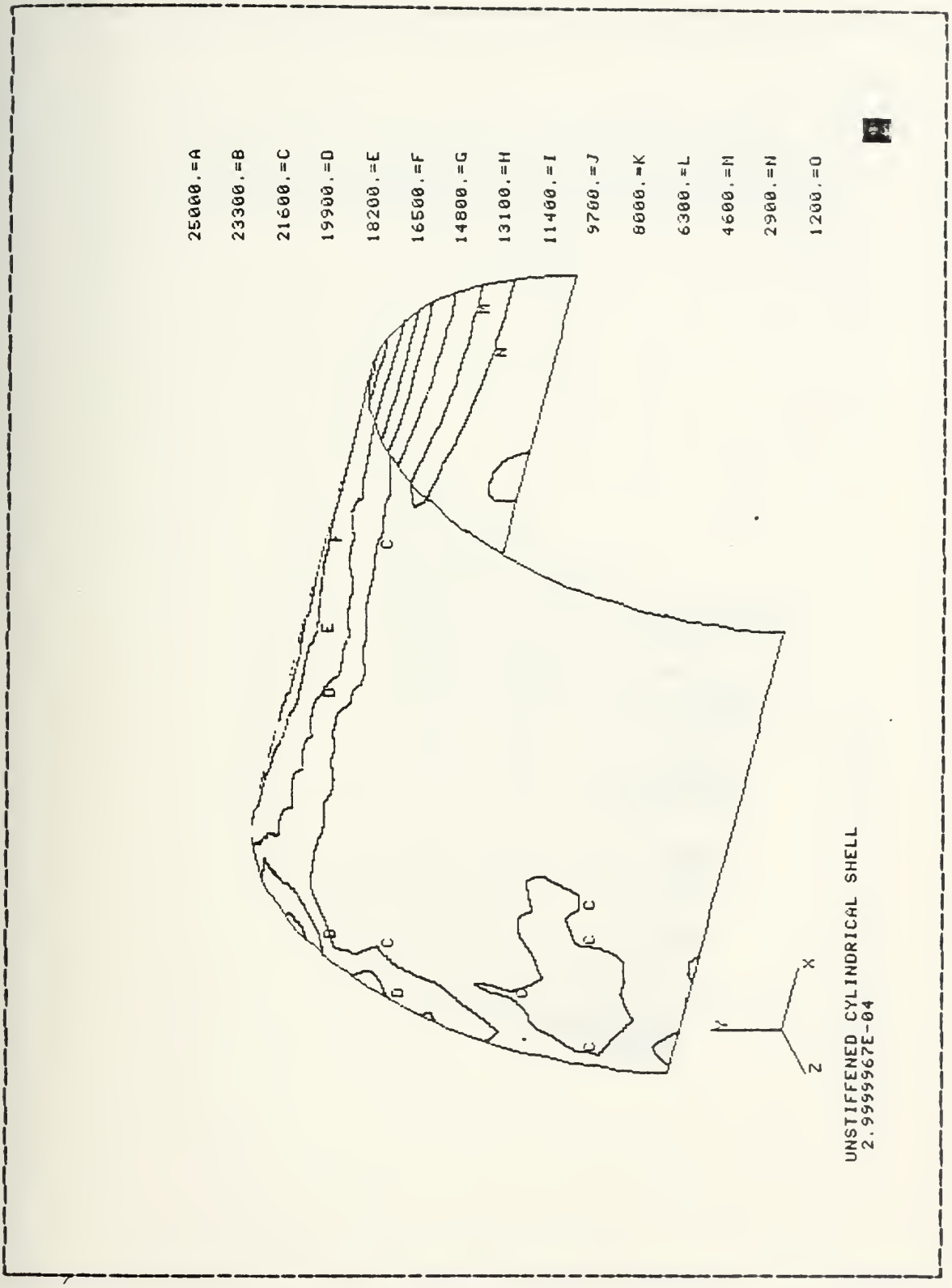


Figure A.17 Von Mises stresses, step 100, unstif. shell.

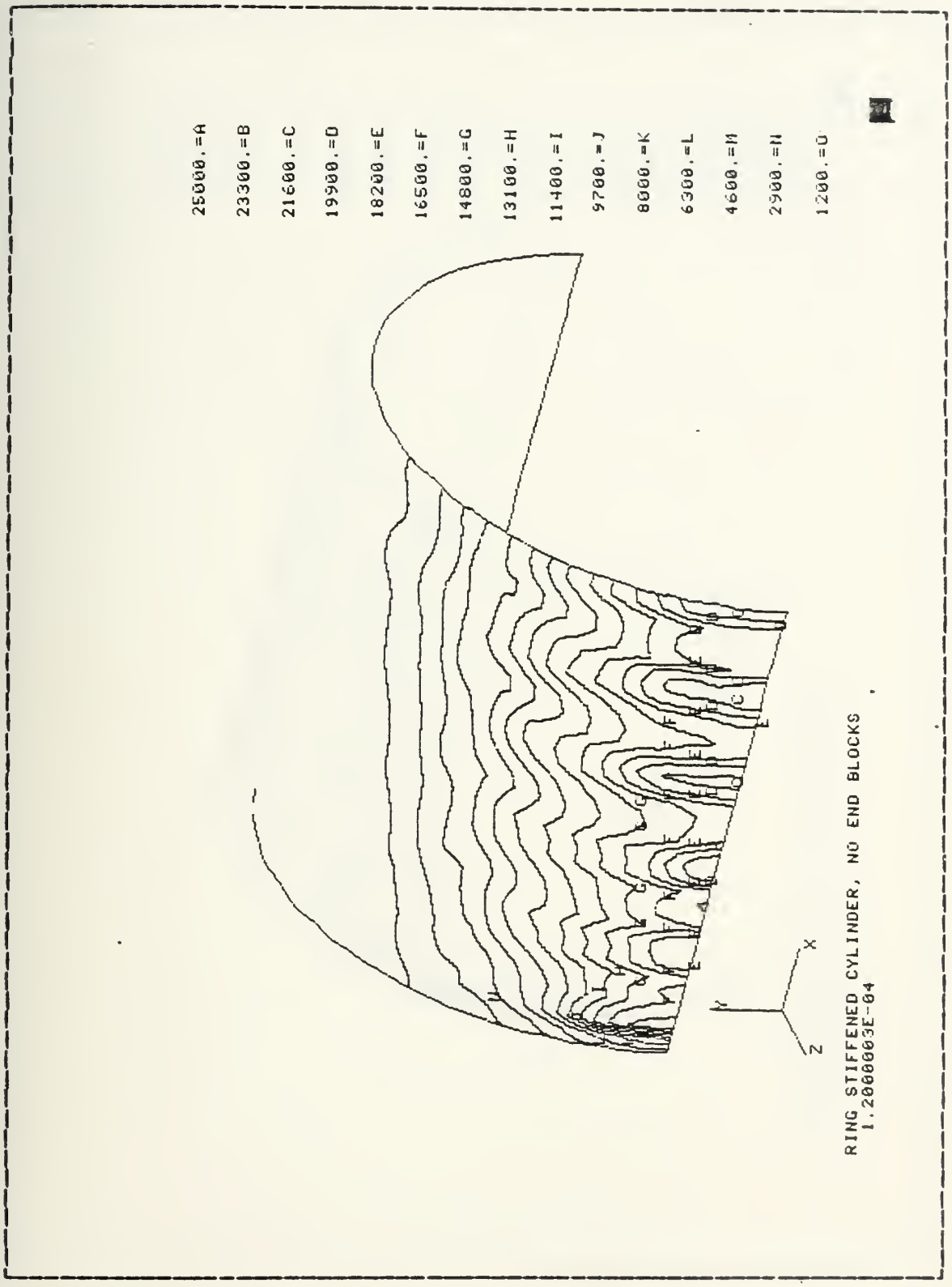


Figure A.18 Von Mises stresses, step 40 , stiff. shell.

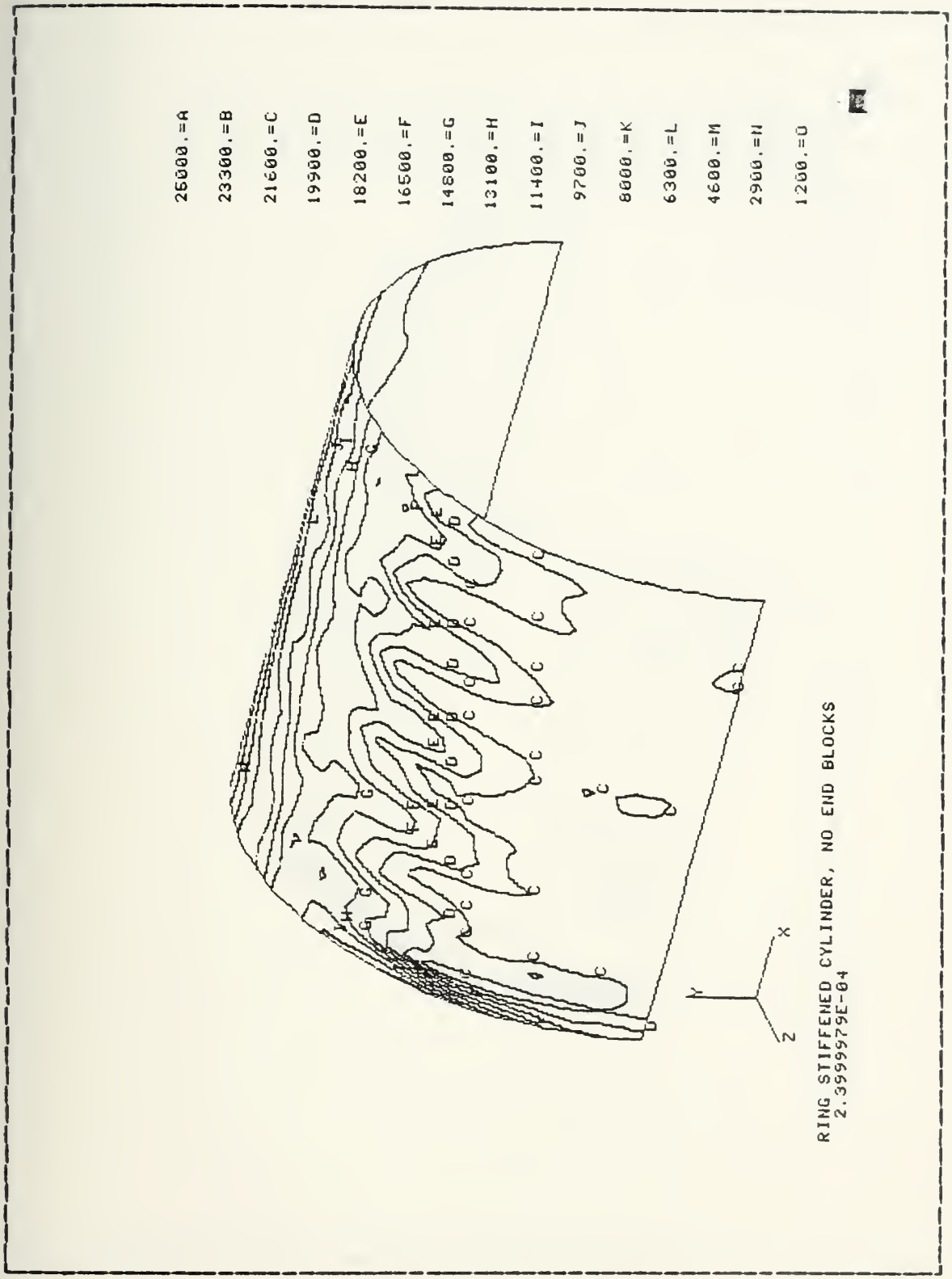


Figure A.19 Von Mises stresses, step 80 , stiff. shell.

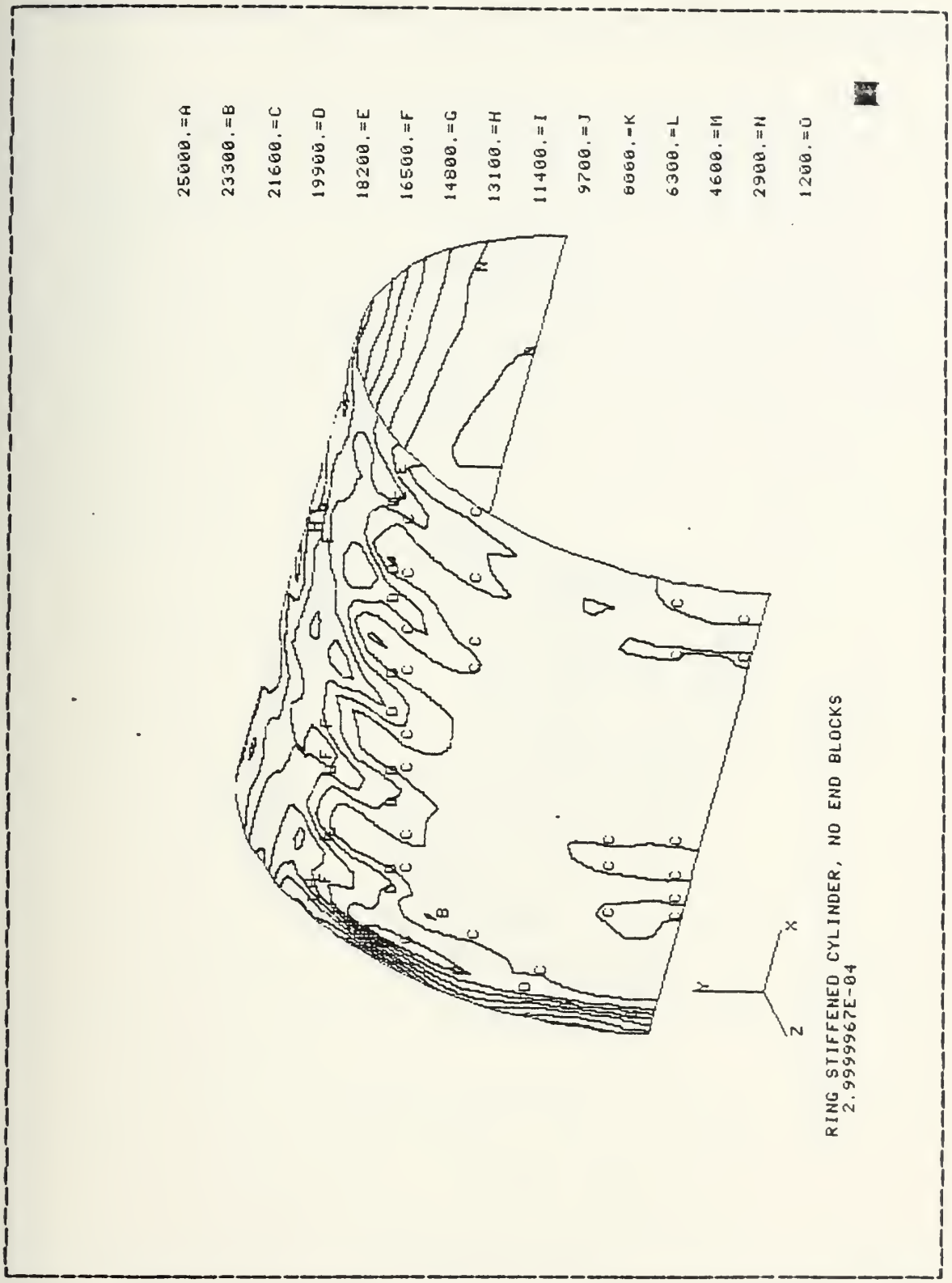


Figure A.20 Von Mises stresses, step 100, stiff. shell.

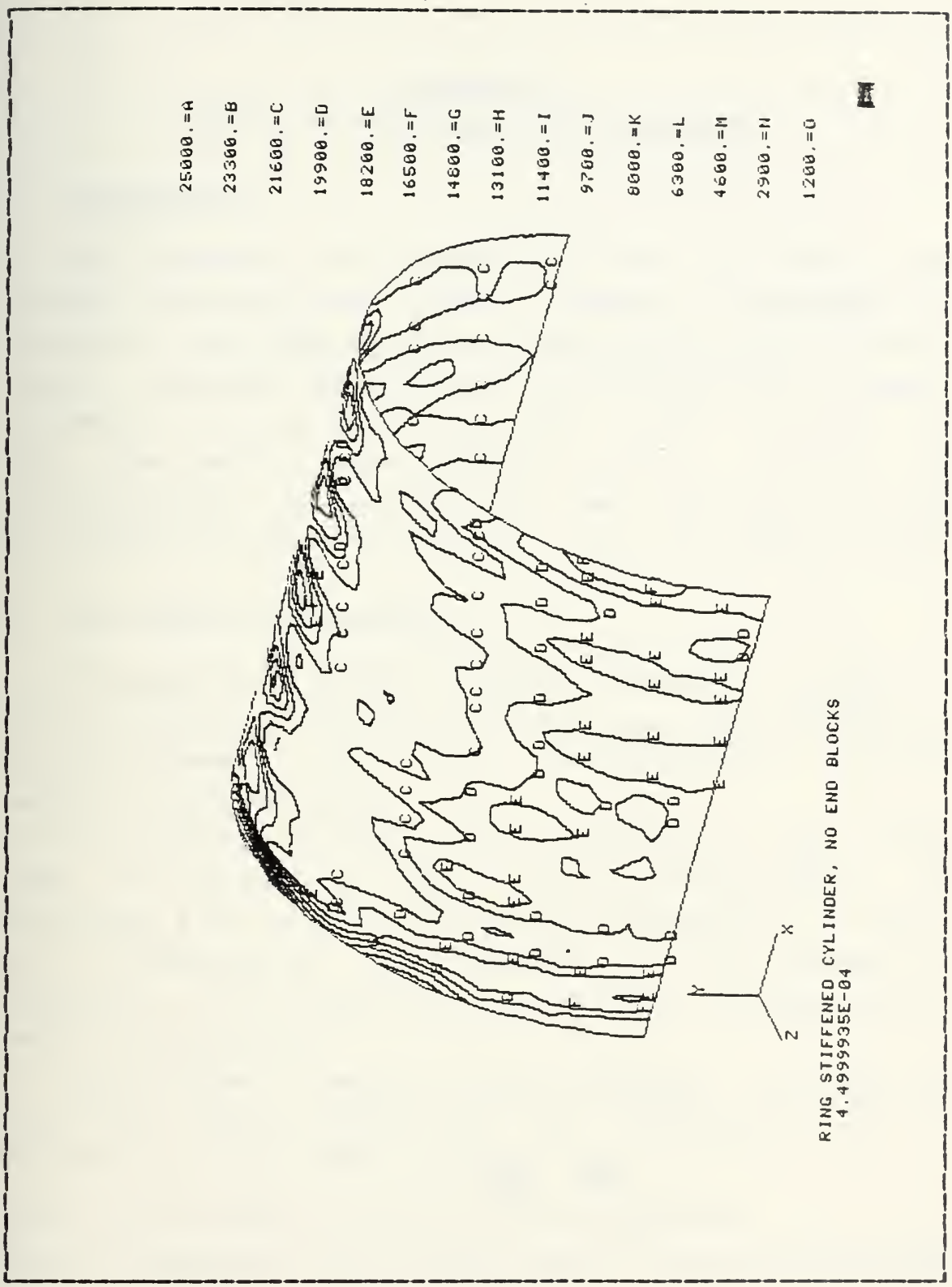


Figure A.21 Von Mises stresses, step 150, stiff. shell.

APPENDIX B
REVIEW OF NONLINEAR FINITE ELEMENTS

A. INTRODUCTION

This appendix is intended to give the reader some insights into nonlinear finite elements. The reader is assumed to have some previous knowledge of finite element theory. The basic principles of the theory will be quickly reviewed, but the study will focus on the problems that occur when dealing with nonlinear theory. Most of the information has been taken from [Ref. 6] as well as from the course the author had at M.I.T. with K.J. Bathe in 1982.

B. THE NEED FOR A NEW THEORY

Considering a coordinate frame defined by (i,j,k), a body of volume V in which point $A(x_1,x_2,x_3)$ is subjected to the displacements (u_1,u_2,u_3), corresponding to a strain vector {e} (figure B.1).
In the following sections, the superscripts 0 and t will refer to the body at time 0 and t respectively, the subscripts 0 and t will refer to the configuration at time 0 and t respectively. This chapter, for the purpose of simplicity, will first deal with the static nonlinear analysis of the material.

In the linear theory of finite elements, one uses the well known Cauchy stress tensor τ_{ij} associated with the engineering strain tensor $e_{ij} = \frac{1}{2}\left[\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right]$.
Then, the principle of virtual work is written:
where ${}^tR$ represents the virtual work of externally applied forces and δe is a virtual strain.

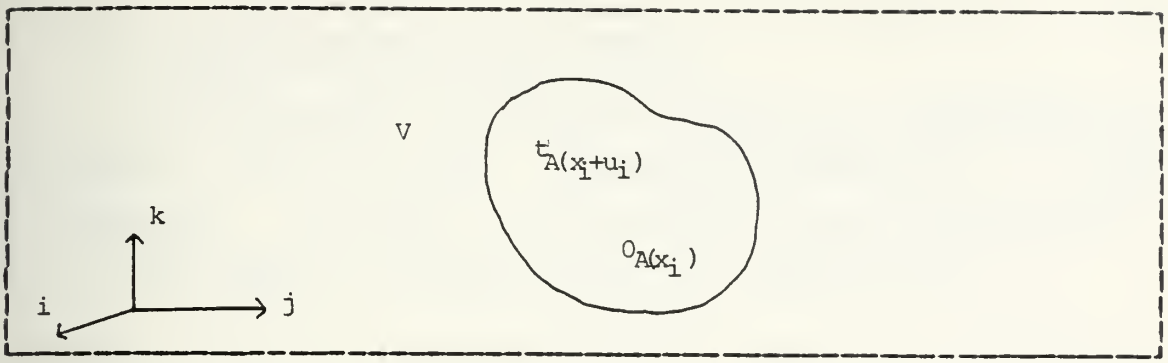


Figure B.1 Geometric Conventions.

$$\int_{^tV} \tau_{mn} \delta e_{mn} = {}^tR \qquad (B.1)$$

Equation (B.1) is then discretized over the body and becomes a set of integrations over each of the finite elements. In the case of a large displacement, the volume of the body over which the integration is performed might have significantly changed. Also notice that equation (B.1) is written in the original coordinate frame (defined at t=0) and that τ_{mn} and e_{mn} refer to the current configuration of the body. The Cauchy stresses at time $t+\Delta t$ cannot be obtained by adding an increment due to the straining of the material to the stresses at time t. The rigid body rotation of the material has to be taken into account since the Cauchy stresses vary under rigid body motions. Therefore, we must perform the integration of equation (B.1) over the unknown current volume with respect to the original geometry that could be significantly different from the current one.

The above discussion emphasizes the need for a new set of stress and strain tensors that would alleviate the aforementioned problems and enable the integration of the principle of virtual work to be performed.

C. DEFINING NEW STRESS AND STRAIN TENSORS

1. Green-Lagrange Strain Tensor

The structure introduced earlier is considered and two points A and B at t=0, of coordinates ${}^0A({}^0x_i)$, ${}^0B({}^0x_i)$ are defined . At time t, the body has been deformed and A and B have moved to ${}^tA({}^tx_i)$ and ${}^tB({}^tx_i)$ (figure B.2).

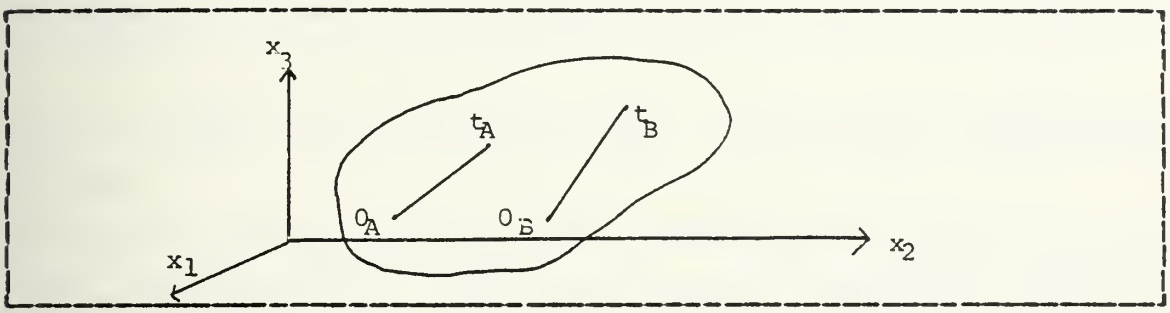


Figure B.2 Displacements Conventions.

A Taylor expansion is used to express the coordinates of B as a function of the coordinates of A.

$$ {}^tx_i = {}^tx_i + \frac{\partial {}^tx_i}{\partial {}^0x_j}({}^0x_j - {}^0x_j) \qquad (B.2) $$

or with $d\,{}^tx_i = {}^tx_i - {}^tx_i$ and $d\,{}^0x_i = {}^0x_i - {}^0x_i$, gives :

$$ (d\,{}^tx_i) = \left(\frac{\partial {}^tx_i}{\partial {}^0x_j}\right)(d\,{}^0x_i) \qquad (B.3) $$

$$ \{d^tx\} = [\,{}^t_0X\,]\,\{d^0x\} \qquad (B.4) $$

where $[\,{}^t_0X\,] = \left(\frac{\partial {}^tx_i}{\partial {}^0x_j}\right)$, $\{d^tx\} = (d\,{}^tx_i)$, $\{d^0x\} = (d\,{}^0x_i)$

In other words, equation (B.4) expresses how a small fiber defined by the vector $\{d\,{}^0x\}$ at t=0 has rotated and

extended between time 0 and time t when it becomes $\{d^t x\}$.
The matrix $[{}^t_0 X]$ is called the "deformation gradient" .
The new length ${}^t dS$ of the fiber will be :

$({}^t dS)^2 = \{d^t x\}^T \{d^t x\}$ (T refers to the transposed matrix)

and therefore

$({}^t dS)^2 = ([{}^t_0 X]\{d^0 x\})^T ([{}^t_0 X]\{d^0 x\})$

$$({}^t dS)^2 = \{d^0 x\} [{}^t_0 C] \{d^0 x\} \qquad (B.5)$$

$[{}^t_0 C] = [{}^t_0 X]^T [{}^t_0 X]$ is called the "Cauchy-Green deformation tensor".

Notice that if the Cauchy-Green deformation tensor is identity, equation (B.5) indicates that the length of any fiber will not vary. In other words, whenever rigid body motions are considered, the Cauchy Green deformation tensor is identity since the fibers do not stretch.

The principles of the finite element method will now be quickly recalled. Assume that the displacement (u_i) of any point of the body can be written as a combination of the displacements of N selected points called "nodes" :

$$(u_i) = \sum_{k=1}^{N} h_k (u_i)^k \qquad (B.6)$$

Where the h are interpolation functions that depend only on the geometry of the body. In addition, the nodes are chosen so as to get a division into quadrilateral elements and it is assumed that the displacement of any point is only a function of the displacements of the corner nodes of the element it belongs to. Then equation (B.6) becomes

$$(u_i) = \sum_{k=1}^{4} h_k (u_i)^k \qquad (B.7)$$

Recalling that (u_i) is simply $({}^t x_i) - ({}^0 x_i)$ gives:

$$({}^{t}x)_i = ({}^{0}x)_i + \sum_{k=1}^{4} h_k (u_i)^k \qquad (B.8)$$

The deformation matrix $[{}^{t}_{0}X]$ can be expressed very simply using the previous equation :

$$[{}^{t}_{0}X] = \left(\frac{\partial {}^{t}x_i}{\partial x_j}\right) = \delta_{ij} + \sum_{k=1}^{4} \frac{\partial h_k}{\partial x_j} (u_i)^k \qquad (B.9)$$

Define now the "Green Lagrange" strain tensor by :

$$[{}^{t}_{0}\varepsilon] = 1/2 \quad ([{}^{t}_{0}C] - [I]) \qquad (B.10)$$

Where [I] is the identity matrix.
From the previous derivations, it can be observed that the Green Lagrange strain tensor is 0 for rigid body motions. The Green-Lagrange strain tensor refers to the body at time t with respect to the initial configuration. This is why it will be so useful in dealing with large deformations.

Recall that the ultimate goal is to apply the principle of virtual work to the structure under study. In particular, having defined a new strain tensor, the relation giving the virtual Green-Lagrange strain tensor corresponding to a virtual displacement $(\delta u)_i$ must be known. In the case of a linear problem, the virtual engineering strain tensor would be:

$$\delta_t e_{ij} = 1/2 \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$$

In the case of large displacements, the Green-Lagrange strain tensor should be used. It is shown in [**Ref. 6**] that the virtual Green Lagrange strain tensor corresponding to virtual engineering strains is:

$$\delta\, {}^{t}_{0}\varepsilon_{ij} = \frac{\partial {}^{t}x_m}{\partial x_i} \frac{\partial {}^{t}x_n}{\partial x_j} \delta t e_{mn} \qquad (B.11)$$

or:

$$\delta_t e_{ij} = \frac{\partial x_i}{\partial x_m} \frac{\partial x_j}{\partial x_n} \delta_0 \varepsilon_{ij} \qquad (B.12)$$

Having defined a strain tensor which is invariant under rigid body motions, a stress tensor corresponding to the Green-Lagrange strain tensor needs to be defined.

2. Stress Measures

Starting with the Cauchy stress tensor , the Piola-Kirchoff stress tensor is defined :

$$ {}^t_0 S_{ij} = \frac{{}^0\rho}{{}^t\rho} {}^0_t x_{im} {}^t\tau_{mn} {}^0_t x_{jn} \qquad (B.13)$$

Where ${}^0\rho$, ${}^t\rho$ are the densities of the material at time 0 and t respectively and $({}^0_t x_{ij}) = [{}^0_t X]^1$ is the inverse of the deformation tensor defined previously.
Equation B.13 can be easily rewritten in equivalent form:

$$ {}^t\tau_{mn} = \frac{{}^t\rho}{{}^0\rho} {}^t_0 x_{im} {}^t_0 S_{ij} {}^0_t x_{jn} \qquad (B.14)$$

It can also be shown by using the principle of mass conservation that :

$$ {}^0\rho = {}^t\rho \det[{}^t_0 X] \qquad (B.15)$$

D. PRINCIPLE OF VIRTUAL WORK

The principle of virtual work of equation (B.1) is rewritten using the strain and stress tensors defined previously in equations (B.12) and (B.14) :

$$ {}^t R = \int_{{}^t V} \frac{{}^t\rho}{{}^0\rho} {}^t_0 x \; {}^t_0 S_{lk} {}^t_0 x_{nk} \; {}^0_t x_{im} {}^0_t x_{jn} \delta {}^t_0 \varepsilon_{ij} \; dV \qquad (B.16)$$

or:

$$ {}^{t}R = \int_{{}^{t}V} \frac{{}^{t}\rho}{{}_{0}\rho} \; {}^{t}_{0}S_{ij} \, \delta \, {}^{t}_{0}\varepsilon_{ij} \, dV \qquad (B.17) $$

and using equation (B.15) :

$$ {}^{t}R = \int_{{}_{0}V} {}^{t}_{0}S_{ij} \, \delta \, {}^{t}_{0}\varepsilon_{ij} \, dV \qquad (B.18) $$

Thus, the principle of virtual work has been simply expressed in terms of a new set of stress and strain tensors, integrated over the original volume ${}^{0}V$.

E. THE INCREMENTAL CONTINUUM MECHANICS EQUATIONS

In this section, the principle of virtual work will be applied to the structure and the incremental formulation using the Piola-Kirchoff and Green-Lagrange tensors will be developed . Non linear terms will arise from the rather complicated definition of the strain tensor, but it must be pointed out that the new formulation provide the means for the modelling of large deformations.

Assume that the configuration of the body at time t is known, the configuration at time $t+\Delta t$ must be determined. Writing the principle of virtual work at time $t+\Delta t$ gives :

$$ \int_{{}_{0}V} {}^{t+\Delta t}_{0}S_{ij} \, \delta \, {}^{t+\Delta t}_{0}\varepsilon_{ij} \, dV = {}^{t+\Delta t}R \qquad (B.19) $$

${}^{t+\Delta t}R$: external work at time $t+\Delta t$

$\delta \, {}^{t,\Delta t}_{0}\varepsilon_{ij}$: virtual increment in G.L. strain

${}^{t+\Delta t}_{0}S_{ij}$: stress state at time $t+\Delta t$

Separating between the known terms that refer to the configuration at time t and the unknown terms that are the increments of stress and strain between time t and time $t+\Delta t$ gives :

$$ {}^{t+\Delta t}_{0}S = {}^{t}_{0}S_{ij} + {}_{0}S_{ij} \quad \text{and} \quad {}^{t+\Delta t}\varepsilon = {}^{t}_{0}\varepsilon_{ij} + {}_{0}\varepsilon_{ij} \quad \varepsilon \qquad (B.20) $$

${}_{0}S_{ij}$ and ${}_{0}\varepsilon_{ij}$ are simply the increments in stress and strain respectively.

It is derived in [Ref. 6] that the increment in Green-Lagrange strain ${}_{0}\varepsilon_{ij}$ is made of a linear part ${}_{o}e_{ij}$ and a nonlinear one ${}_{0}\eta_{ij}$. The term linear refers to the increment in displacement u_i.

$$ {}_{0}\varepsilon_{ij} = {}_{o}e_{ij} + {}_{0}\eta_{ij} \quad \text{or} \quad \delta\, {}_{0}\varepsilon_{ij} = \delta\,({}_{o}e_{ij} + {}_{0}\eta_{ij}) \qquad (B.21) $$

Using equation (B.21) in equation (B.19) gives :

$$ \int_{0_V} ({}^{t}_{0}S_{ij} + {}_{0}S_{ij}) (\delta\, ({}_{0}e_{ij} + {}_{0}\eta_{ij})) \, dV = {}^{t+\Delta t}R \qquad (B.22) $$

Again, separating between the known and unknown terms gives :

$$ \int_{0_V} {}^{t}_{0}S_{ij}\, \delta\, {}^{t}\varepsilon_{ij}\, dV + \int_{0_V} {}^{t}_{0}S\, \delta\, {}_{0}\eta_{ij} = {}^{t+\Delta t}_{R} - \int_{0_V} {}^{t}_{0}S_{ij}\, \delta\, {}_{0}e_{ij} \qquad (B.23) $$

For small increments in displacements, equation (B.23) written at time t indicates that $\delta\, {}_{0}e_{ij} = \delta\, {}_{0}\varepsilon_{ij}$. This signifies that in equation (B.21) the non linear term is negligible

Then the constitutive law of the material detailed in chapter II allows to relate stresses and strains:

$$ {}_{0}S_{ij} = {}_{0}C_{ijrs}\, {}_{0}\varepsilon_{ij} \simeq {}_{0}C_{ijrs}\, {}_{0}e_{ij} \qquad (B.24) $$

and B.23 becomes:

$$ \int_{0_V} \delta\, {}_{0}e_{ij}\, {}_{0}C_{ijrs}\, {}_{0}e_{rs}\, dV + \int_{0_V} {}^{t}_{0}S_{ij}\, \delta\, {}_{0}\eta_{ij}\, dV = {}^{t+\Delta t}R - \int_{0_V} {}_{0}S_{ij}\; {}_{0}e_{ij} dV \qquad (B.25) $$

The right hand side terms of the previous equation are known:

${}_0\delta e_{ij}$ is the linear increment in virtual strain involving only known terms

${}_0^t S_{ij}$ is known from the previous time step

${}^{t+\Delta t}R$ is the work of external prescribed virtual forces.

The left hand side of B.25 is unknown since ${}_0 e_{ij}$ and ${}_0 \eta_{ij}$ involve the displacement from time t to time $t+\Delta t$.

F. FINITE ELEMENT DISCRETIZATION

Equation (B.25) will be discretized over the structure, using the finite element approximation defined in section B.9 . Let N be the number of nodes, the principle of virtual work will be invoked N times, setting a unit displacement at each node in turn:

$$\delta u_k = 1 \quad , \quad \delta u_j = 0 \quad , k \neq j$$

A system of equations whose unknowns are the nodal displacements is obtained Let $\{\Delta u\}$ be the vector of unknown displacements, $\{F\}$ be the vector of nodal point forces equivalent to the internal stresses.

Then equation (B.25) can be rewritten in matrix form :

$$[{}_0^t K_L]\{\Delta u\} + [{}^t K_{NL}]\{\Delta u\} = \{{}^{t+\Delta t}R\} - \{{}^{t+\Delta t}F\} \qquad (B.26)$$

$[{}_0^t K_L]$ and $[{}_0^t K_{NL}]$ are known from the material characteristics at time t and correspond respectively to a linear and nonlinear contribution. It is therefore possible to solve equation (B.25) for $\{\Delta u\}$. However, because of the assumption in equation (B.24), the exact solution might not be reached immediately. Furthermore, depending on the time step size, the solution process might even be unstable! In any case,

an iteration for solving equation (B.26) must be performed until the exact solution is reached .
A widely used scheme is the "modified Newton iteration" defined by the following equation and boundary conditions :

$$\left([{}_0^tK_L] + [{}_0^tK_{NL}] \right) \{\Delta u\}^i = {}^{t+\Delta t}\{R\} - {}^{t+\Delta t}\{F\}^{i-1} \qquad \text{(B.27)}$$

and $\{{}^{t+\Delta t}u\}^i = \{{}^{t+\Delta t}u\}^{i-1} + \{\Delta u\}^i$, with the initial conditions :
$\{{}^{t+\Delta t}u\}^0 = \{{}^tu\}$ and $\{{}^{t+\Delta t}F\}^0 = \{{}^tF\}$

$\{\Delta u\}^i$ is the vector of incremental nodal point displacements at iteration i.
${}^{t+\Delta t}\{R\}$ is the vector of applied loads (constant in the iteration)
${}^{t+\Delta t}\{F\}^{i-1}$ is the vector of nodal point forces equivalent to the stresses at time $t+\Delta t$, iteration i-1 .

At the first iteration, equation (B.27) reduces to equation (B.26) giving an increment of displacement $\{\Delta u\}^1$. Then a better approximation of ${}_0e_{rs}$, $\delta {}_0\tau_{ij}$ is obtained . ${}_0^tS_{ij}$ is updated to the new state of stresses and becomes ${}_0^{t+\Delta t}S_{ij}$. Equation (B.27) is then used to determine the new increment in displacement $\{\Delta u\}^2$, and so on until the increment in displacement is small enough, so that ${}^{t+\Delta t}\{R\} = {}^{t+\Delta t}\{F\}^i$ in equation (B.27)

G. INCLUSION OF DYNAMIC FORCES

If the loads are applied rapidly, inertia forces need to be considered and a truly dynamic problem has to be solved. Using d'Alembert's principle, the element inertia forces are simply included as part of the body forces. Let $\{\ddot{u}\}$ be the vector of nodal accelerations and [M] be the mass matrix of the system. Then the principle of virtual work is written in the following way :

$$([{}^{t}_{0}K_{L}]\{\Delta u\} + [{}^{t}_{0}K_{NL}]\{\Delta u\}) = {}^{t+\Delta t}R - [M]\{\Delta \ddot{u}\} - {}^{t+\Delta t}F \qquad (B.28)$$

Equation (B.28) represents a system of differential equations of second order. If the non linear term $[{}^{t}_{0}K_{NL}]$ were negligible, the solution could be obtained by standard procedures for solving differential equations with constant coefficients. However, the procedures proposed for the solution of general systems of differential equations can become very expensive if the order of the matrices is large. Therefore, whenever the system is linear or nonlinear, some effective methods for solving the equations governing the equilibrium are required.

1. Direct Integration Methods

The essence of direct integration methods is based on two ideas. First, it is aimed to satisfy B.28 only at certain time intervals apart. Second, a variation of acceleration velocities and displacements is assumed within each time step. The form of the assumption determines the accuracy, stability and cost of each method.

In the following, assume that the initial conditions (displacements, accelerations, velocities) at time 0, denoted (${}^{0}u$, ${}^{0}\dot{u}$, ${}^{0}\ddot{u}$) are known. In the solution, the time span under consideration, T , is subdivided into n equal time intervals Δt. Assuming that the solution is known at time t , the methods of getting the solution at time $t+\Delta t$ will be investigated.

2. Central Difference Method

In the Central Difference method, a finite difference approximation will give the acceleration at time t :

$${}^{t}\ddot{u} = \frac{1}{\Delta t^{2}}({}^{t-\Delta t}u - 2\,{}^{t}u + {}^{t+\Delta t}u) \qquad (B.29)$$

Writing the principle of virtual work at time t and substituting into equation (B.30) gives then:

$$[M]\{{}^{t}\ddot{u}\} + [{}^{t}_{0}K]\{{}^{t}u\} = \{{}^{t}R\} \qquad (B.30)$$

$$\frac{1}{\Delta t^2}[M]\{{}^{t+\Delta t}u\} = \{{}^{t}R\} - ([{}^{t}_{0}K] - 2[M]/\Delta t^2)\{{}^{t}u\} - \frac{1}{\Delta t^2}[M]\{{}^{t-\Delta t}u\} \qquad (B.31)$$

The previous equation gives the deformation at time $t+\Delta t$ from the characteristics of the system at time t .

When [M] is diagonal, which is frequently the case for mass matrices, the solution at time $t+\Delta t$ does not involve any triangular factorization of the matrix [M] , thus leading to more efficient computations.

The shortcoming in the use of the central difference method lies in the time step restriction: for stability, the time step size t must be smaller than a critical time step Δt_{cr} which is equal to T/π , where T is the smallest period of the finite element system.

The central difference scheme is fairly easy to implement for the integration of a system of non linear differential equations. However, because of the limitations of the time step , it might not be suitable for cases when loads are varying at a slow pace.

3. Implicit Integration Schemes

Since the interest of this study lies in central difference schemes, this section will be limited to a short description of the fundamentals of implicit time integration schemes.

Implicit time integration schemes use the principle of virtual work written at time $t+\Delta t$ and not at time t as for the central difference method :

$$[M]\,{}^{t+\Delta t}\{\ddot{u}\} + {}^{t+\Delta t}_{0}[K]\,{}^{t+\Delta t}\{u\} = {}^{t+\Delta t}\{R\} \qquad (B.32)$$

Again, using a finite difference approximation of ${}^{t+\Delta t}\{u\}$ and replacing into equation (B.32) enables to solve for ${}^{t+\Delta t}\{\ddot{u}\}$. Since the formulation involves the rigidity matrix ${}^{t+\Delta t}_{0}[K]$ and the external work ${}^{t+\Delta t}\{R\}$ which are both unknown, the system has to be solved in a similar way to the Modified Newton iteration that was detailed previously.

Implicit time integration schemes are stable regardless of the size of the time step used. However, if the time step size is too large, significant errors can be accumulated at each time step, leading to unrealistic results.

The reader will find more details on the various implicit method in [Ref. 6]. Yet, it can pointed out that implicit methods are more tedious to implement. On the other hand, a larger time step can be used in the solution procedure, which can be of extreme importance when studying phenomena over a significant period of time.

APPENDIX C

HOW TO USE THE TRANSLATOR EPSA-PATRAN-G

This appendix is intended to explain in detail the use of EPSA/PATRAN-G post-processing facilities. It is divided in three sections that will deal successively with 1) the displaying of the original model ; 2) the deformed geometry; 3) the contour plots of element and nodal points quantities.

A. DISPLAYING THE ORIGINAL MODEL

When making an initial EPSA analysis on a particular structure, the geometry of the model has to be input into PATRAN-G. As explained in chapter III, all the geometrical information is contained in a file FOR019.DAT that is created each time an EPSA run is made. The input of the original geometry must be made via the neutral input mode of EPSA. The procedure, starting from the "logon" to PATRAN-G is the following :

- Select the GO option
- Select the new data file option (option 1)
- Select the neutral system (option 4)
- Select the input mode (option 4)
- Input the neutal file name : FOR019.DAT

The original geometry will then be displayed on the screen

It is often found convenient to have a perpective view of the model under study. In that case, the user should :

- Issue the VIEW command
- Select the rotation about the absolute axes (option 1)
- Input an angle of rotation

(23,-34,0 will give a very nice view but any angle can be input)

- Issue a PLOT command to have the model displayed in the new axes.

An example of the procedure is provided on table VI.

TABLE VI

Finite Element Model Input Procedure

```
MODE?  1.GEOMETRY MODEL 2.ANALYSIS MODEL 3.DISPLAY 4.NEUTRAL SYS. 5.END
>4
NEUTRAL FILE?  1.CREATE OUTPUT  2.INPUT MODEL  3.POST-PROCESSING  4.END
>2
INPUT NEUTRAL FILE NAME
>FOR019.DAT
DO YOU WISH TO OFFSET ANY NEUTRAL INPUT IDS? (Y/N)
>N
 EPM 200 STEPS ,NO STIFFENERS W=30.
 SHALL WE PROCEED WITH THE READING OF THIS FILE? (Y/N)
>
```

The PLOT command can be issued anytime to display the original geometry on the screen.

When studying ccmplex models, one does not want the element and node numbers to be printed along with the geometry of the structure. The command SET, LABS3, OFF followed by a PLOT will display the original geometry without any labels printed.

When a model has been input into PATRAN-G, a data file PATRAN.DAT is created on the user's directory. When connecting with PATRAN-G at a later time, the user can

select the option "last data file " (option3) to have the original geometry displayed on the screen without having to input the model again.

B. DEFORMED GEOMETRY

On the second card of the PATRAN-G input deck, the user specifies the number of PATRAN-G displacement data file and the number of element results data file. Two non-zero integers in free format must be placed at the end of the second card (section II in the user's manual) requesting the number of displacements and of elements files respectively.

Assuming that the user has made a 200 steps run with 10 output requests for PATRAN-G displacement files, ten (10) files FOR018.DAT will then be created at equal time intervals. The deformed geometry corresponding to time step 100 will therefore be contained in FOR018.DAT;5.

To display the deformed geometry corresponding to time step 100, the user should issue the following commands:

- RUN,DEF : requests deformed geometry option
- Input the name of the displacements file :FOR018.DAT 5
- Select the PLOT option (option 3)
- Select the undeformed geometry (2) followed by the deformed geometry (3). An example of the procedure is provided on table VII. The undeformed geometry superposed with the deformed geometry will then be displayed on the screen.

C. POST-PROCESSING OF ANALYIS RESULTS.

Element-related quantities like von Mises stresses are contained in FOR016.DAT files, nodal point quantities are stored in FOR018.DAT files. As described in chapter III, each column of those files contains a specific quantity

TABLE VII

Deformed Geometry Procedure

```
>
MODE?  1.GEOMETRY MODEL 2.ANALYSIS MODEL 3.DISPLAY 4.NEUTRAL SYS. 5.END
>RUN,CON,COL,4
 CURRENT FILE FOR NODAL RESULTS IS   FOR018.DAT
NEUTRAL RESULTS FILE?  1.NEW FILE  2.CURRENT FILE
>1
INPUT THE RESULTS FILE NAME:
>FOR018.DAT;5
 DATA WIDTH =     5
 FILE TITLE =EPM 800 STEPS
              7.5000129E-04


 DATA VALUES RANGE FROM -0.525E+00 TO  0.134E+00
ASSIGNMENT?  1.AUTO  2.MANUAL  3.SEMI-AUTO  4.USE CURRENT LEVELS  5.END
>4
 PREVIOUS CONTOUR LEVELS USED.
MODE?  1.GEOMETRY MODEL 2.ANALYSIS MODEL 3.DISPLAY 4.NEUTRAL SYS. 5.END
>RUN,HI,C
```

(i.e. column 31 of FOR016.DAT contains the von Mises stresses). The reader will refer to chapter III for the detailed organization of those files.

Again, assume a 200 time steps analysis, with 10 output requests for element results files. The user might want to display the contour plots of von Mises stresses at time step 100 . In this case the following commands should be input :

- RUN, CON, COL, 31: tells PATRAN-G to look at column 31 that contains the von Mises stresses.
- Input the file name FOR016.DAT 5
- PATRAN-G will then ask for a color assignment (automatic, manual, semi-automatic, current levels used) that the user will select according to his needs.

The contour plots are then ready to be displayed :

- RUN, HI, FR would display the color-filled contour plots
- RUN, HI, CON would display the contour plots (color lines)

The von Mises stresses are the most useful element quantities to be displayed, but other element-related quantities detailed in chapter III like the x and y stresses in local or global coordinates could be displayed as well by looking at their corresponding column in the element results data file.

Dealing with nodal point quantities, the displacement normal and the velocity normal to the shell are very meaningful quantities in an analysis. They are respectively stored in column 4 and 5 of the displacement results files FOR018.DAT .
A contour plot of the normal displacement at time step 100 would then be obtained via the following commands:

- RUN, CON, COL,4 (look at column 4)
- FOR018.DAT 5 (name of the file)
- Color assignment chosen
- RUN, HI, C or RUN, HI, FR

Notice When the fluid-structure interaction is ON, the normal displacement contained in column 4 corresponds to pure deformations, the rigid body contribution having been taken out.

All the element results processing is implemented in the routine NEUSTRE, all the nodal points processing is implemented in routine NEUDISP. It should be pointed out that any modification to the capabilities of the translator (i.e. being able to display other types of quantities) can be made by modifying those routines only.

APPENDIX D
LISTINGS

This appendix contains the various files that constitute the translator module. The submodule that displays the original geometry is listed on the first four pages. It is imbedded in file FRANK.FOR. The submodule that takes care of the post-processing facilities is imbedded in file DISP.FOR and is listed in the remaining pages. It has been mentioned previously that EPSA had been slightly modified to accomodate color graphics capabilities. The only interaction of EPSA with the translator occurs via subroutine calls. All the "calls" occur in COMPUTE (for post-processing) and REPORT (for the original geometry). A labelled COMMON called FRANK has been created and is defined in the routine AAA as required by EPSA. The requests for PATRAN-G outputs are echoed in the EPSA output file, all the modifications for that purpose having been made in routine READIN. The user must be aware that the size of blank COMMON array A has been increased to store the deflections in x, y, z directions instead of only the z direction previously.

```
      subroutine prelim
      dimension A(1)
      common ia(1)
      equivalence (ia(1),a(1))
      common/ssize/ ibq(1),nqj,neltot,nlbd,nload,nbrect,
     1  nbquad, isheet, norpts, nsots, nstrpts, nvpts, nhpts,
     2  nsstyp, nnj, nntot, lqdsp, liqud, lbcalc
c     common /cpara/ nsteps, nbeg, nend, nsheet
      common /stab / ibt(1),jssize,jspar,jvelo,jstre,jxmas,jielm,jbmat,
     1    jllbd,jloqp,jpret,jlhis,jstrn,jforc,jxloc,jnqi,jnni,jnqbeg,
     2    jlside
c
      common/frank/nfntot,llu
      COMMON/TITRE/NTITLE(80)
      CHARACTER*9 BUFF
      CHARACTER*5 TITLE
      CHARACTER*8 TIM
      CHARACTER*9 VER
c
c        this routine computes new number of nodal points
c         for any given sheet
c
      KJ=JNQBEG-JNNI
      nfntot=nntot-IA(JNNI)-IA(JNQBEG-1)-2*(KJ-2)
      lltype=25
      llkc=1
      lliv=0
      llid=0
      lln1=0
      lln2=0
      lln3=0
      lln4=0
      lln5=0
      write(llu,10) lltype,llid,lliv,llkc,lln1,lln2,lln3,lln4,lln5
 10   format(i2,8i8)
      write(llu,11) (NTITLE(I),I=1,80)
 11   format(80 A1)
c
c     taking care of second packet
c
      lltype=26
      llkc=1
      lln1=nfntot
      lln2=neltot
      write(llu,10) lltype,llid,lliv,llkc,lln1,lln2,lln3,lln4,lln5
      call date(BUFF)
      write(llu,12) BUFF
 12   format(A,3X,'17:12:09',5X,'1.4')
      return
      end



      subroutine shconn
      dimension A(1)
      common ia(1)
      equivalence (ia(1),a(1))
      common/ssize/ ibq(1),nqj,neltot,nlbd,nload,nbrect,
     1  nbquad, isheet, norpts, nsots, nstrpts, nvpts, nhpts,
     2  nsstyp, nnj, nntot, lqdsp, liqud, lbcalc
```

```
c      common /cpara/ nsteps, npeg, nend, nsheet
       common /stab / ibt(1),jssize,jsoar,jvelo,jstre,jxmas,jielm,jbmat,
      1     jllod,jlodp,jpret,jlhis,jstrn,iforc,jxloc,jngi,jnni,jngbeg,
      2     jlside
c
       common/frank/nfntot,llu
c
c       this routine will write the element corner nodes of each
c       element on neutral file
c
       lltype=2
       llkc=2
       llnod=4
       lliv=4
       LLN1=0
       LLN2=0
       LLN3=0
       LLN4=0
       LLN5=0
       LLCONF=0
       LLPID=0
       LLCEI=0
       THET1=0.
       THET2=0.
       THET3=0.
C
C   INITIALIZATION DONE
C
       norev=0
       llel=jnni-jngi
c
       do 200 k=1, llel
       llrow=IA(jngi+k-1)
c   number of elts in each row
            do 100 j=1,llrow
            lel=IA(jngbeg+K-1)+j-1
            nlel1=j+norev
            nlel2=j+1+norev
            nlel3=j+1+norev+llrow+1
            nlel4=j+norev+llrow+1
c ready to display packet
c
            llid=LEL
            write(llu,80) lltype,llid,lliv,llkc,LLN1,LLN2,LLN3,LLN4,LLN5
 80      format(i2,8i8)
            write(llu,81) llnod,LLCONF,LLPID,LLCEI,THET1,THET2,THET3
 81      format(i8,3i8,3e16.9)
            write(llu,82) nlel1,nlel2,nlel3,nlel4
 82      format(10i8)
 100        continue
c
       norev=norev+llrow+1
 200   continue
       return
       end


       subroutine sheetf
       dimension A(1)
       common ia(1)
       equivalence (ia(1),a(1))
```

```
      common/ssize/ ibg(1),nqj,neltot,nlbd,nload,nbrect,
     1  noquad, isheet, norots, nspts, nstrpts, nvpts, nnpts,
     2  nsstyp, nnj, nntot, lgdso, liqud, lbcalc
c     common /cpara/ nsteps, nbeg, nend, nsheet
      common /stab / ibt(1),jssize,jspar,jvelo,jstre,jxmas,jielm,jbmat,
     1    jllbd,jlodo,jpret,jlhis,jstrn,jforc,jxloc,jnqi,jnni,jnqbeg,
     2    jlside
c
      common/frank/nfntot,llu
      character*1 gtype
c
c  renumbers the nodes, change coordinates, display packet 1
c
      lltype=1
      llid=0
      lliv=0
      llkc=2
      lln1=0
      lln2=0
      lln3=0
      lln4=0
      lln5=0
      ncoun=0
      lldof=6
      llicf=1
      gtype='G'
      llconf=0
      llcid=0
      LSPC1=0
      LSPC2=0
      LSPC3=0
      LSPC4=0
      LSPC5=0
      LSPC6=0
C
C   INITIALIZATION DONE
C
      kj=jnqbeg-jnni
      krow=kj-2
      do 200 j=1,krow
c          loop on new nb. of row
c
           ncoun=ncoun+IA(jnni+j-1)
           llot=IA(jnni+j-1)-2
           do 100 l=1,llot
              llid=llid+1
              xlloc=A(ncoun*2+jxloc+2*l)
              ylloc=A(ncoun*2+jxloc+2*l+1)
c  skip first node of row
c
              zlloc=0.
              if (A(jvelo-2).ne.0.) then
              rcourb=1./A(jvelo-2)
              theta=xlloc/rcourb
              xlloc=rcourb*sin(theta)
              zlloc=rcourb*cos(theta)-rcourb
              endif
c
              if (a(jvelo-1).ne.0.) then
              rcourb=1./a(jvelo-1)
              theta=vlloc/rcourb
```

```
               ylloc=rcourb*sin(theta)
               zlloc=rcourb*cos(theta)-rcourb
               endif
c
               if((a(jvelo-1).ne.0.).and.(a(jvelo-2).ne.0.)) then
               write(llu,70)
 70             format('error,two curvatures are non zero')
               endif
c
c       ready to display packet
c
          write(llu,80) lltype,llid,lliv,llkc,lln1,lln2,lln3,lln4,lln5
 80               format(i2,8i8)
               write(llu,81) xlloc,ylloc,zlloc
 81               format(3e16.9)
               write(llu,82) llicf,gtype,lldof,llconf,llcid,LSPC1,LSPC2,
     1  LSPC3 LSPC4,LSPC5,LSPC6
 82               format(I1,a1,3i8,2x,6i1)
 100       continue
 200    continue
       return
       end
c
c
           subroutine endneu
       common/frank/nfntot,llu
       lltype=99
       llid=0
       lliv=0
       llkc=1
       write(llu,80) lltype,llid,lliv,llkc
 80    format(i2,3i8)
       return
       end
```

```
      SUBROUTINE NEUOISP(DEFL,VELO)                                          REPORT
      DIMENSION A(1)                                                BLANC  2
      COMMON IA(1)                                                  BLANC  3
      EQUIVALENCE (IA(1), A(1) )                                    BLANC  4
      DIMENSION DEFL(1),VELO(1)
      COMMON /CPARA/ NSTEPS , NBEG , NENO , NSHEET , N2BO , N38D1 ,      CPARA  2
     1   N3802, INTRVL, OELT, NHTOT, NJOIN, NRELAX, ALPHA               CPARA  3
     $ LEN S8X                                                          MSPARA 3
      COMMON / SSIZE / IBG(1),NQJ,NELTOT,N1BO,NLOAO,NBRECT,             SSIZE  2
     1   NBQUAD, ISHEET, NPRPTS, NSPTS, NSTRPTS, NVPTS, NHPTS,          SSIZE  3
     2   NSSTYP, NNJ, NNTOT, LGOSP, LIQUO, LBCALC                       SSIZE  4
C                                                                       REPORT 8
      COMMON /STAB / IBT(1),JSSIZE,JSPAR,JVELO,JSTRE,JXMAS,JIELM,JBMAT, STAB   2
     1   JL18D,JLOOP,JPRET,JLHIS,JSTRN,JFORC,JXLOC,JNQI,JNNI,JNQBEG,     STAB   3
     $   JLSIDE,JIELMCL,JSTIF,JDEFL,JFORLG,                             STAB   4
     2   JIFPAR,JFLPAR,JXCOORO,JYCOORO,JOELTAX,JOELTAY,JVMA,JSEFX,       STAB   5
     3   JFLUFR, JPRINC, JVELRAD, JGENFR,JPRES,JCSEP                    STAB   6
      COMMON/CIO/ NIN,NOUT,NTHIST,NCORT,MCORT,NTPLOT,NVMA,ITITLE(20)
      COMMON/FRANK/NFNTOT,LLU,LLN,LLS,NOPLTS,NSPLTS
      COMMON/TITRE/NTITLE(80)
      COMMON/COELT/ISTEP,TIME
C
C
C
      INTEGER NSUBT1(80),NSUBT2(80)
C
      KJ=JNQBEG-JNNI
      NFNTOT=NNTOT-IA(JNNI)-IA(JNQBEG-1)-2*(KJ-2)
      LLU=19
      LLN=18
      LLM=17
      DEFMAX=0.
      MAXNOD=NFNTOT
      NWIDTH=5
      NDMAX=0
      LLID=0
      NCOUN=0
      KROW=KJ-2
          DO 10 J=1,80
          NSUBT1(J)=0
 10       NSUBT2(J)=0
C
C PUT THE TIME STEP ON FIRST SUBTITLE:
C
      CLOSE(UNIT=LLM)
      OPEN(UNIT=LLM,STATUS='NEW')
      WRITE(LLM,*) TIME
      CLOSE (UNIT=LLM)
      READ(LLM,99) (NSUBT1(J),J=1,80)
      CLOSE(UNIT=LLM)
      OPEN(UNIT=LLM,STATUS='OLD')
      OPEN(UNIT=LLN,FORM='UNFORMATTED',STATUS='NEW')
C
C     KROW IS THE NEW NUMB. OF ROWS ,KJ IS THE OLD
C     NCOUN COUNTS NODAL PTS IN EPSA, LLID COUNTS FOR ACTUAL MODEL
C     NUMR COUNTS NODAL PTS IN EPSA
C
C  FIRST LOOP TO GET MAX. DEFORMATION AND NODE NUMBER
C
          DO 200 J=1,KROW
```

```
          NCOUN=NCOUN+IA(JNNI+J-1)
          LLPT=IA(JNNI+J)-2
C
C      LLPT IS NUMB. OF POINTS (FOR PATRAN) IN EACH ROW
C      WE TAKE THE RIGID BODY MOTION OUT BY SUBTRACTING THE V AND W
C      DISPLACEMENTS AT ENDBLOCK AT EACH NODAL POINT
C      NCOUN+2 IS THE POINT NB. (EPSA) FOR ENDBLOCK
C
                RBY=DEFL(3*(NCOUN+2)-1)
                RBZ=DEFL(3*(NCOUN+2))
                IF (LIQUD.EQ.0)  THEN
                RBY=0.
                RBZ=0.
                ENDIF
                DO 100 L=1,LLPT
                LLID=LLID+1
                NUMR=NCOUN+L+1
                  DX=DEFL(3*NUMR-2)
                  DY=DEFL(3*NUMR-1)-RBY
                  DZ=DEFL(3*NUMR)-RBZ
C
C                   IS SHELL CURVED,CHANGE COORDINATES
C
                   IF((A(JVELO-1).NE.0.).OR.(A(JVELO-2).NE.0.)) THEN
                   CALL CHCOORD(NUMR,DX,DY,DZ)
                   ENDIF
                  DD=AMAX1(ABS(DX),ABS(DY),ABS(DZ))
                  IF(ABS(DEFMAX).LT.DD) THEN
                  NDMAX=LLID
                  DEFMAX=DD
                  IF(ABS(AMIN1(DX,DY,DZ)).EQ.DD) THEN
                       DEFMAX=-DEFMAX
                       ENDIF
C
C                 WE CHECKED  IF DEFMAX WAS NEG.
C
                  ENDIF
 100            CONTINUE
 200         CONTINUE
C          OK FOR FIRST CARD
       WRITE(LLN) (NTITLE(I),I=1,80),NFNTOT,MAXNOD,DEFMAX,NDMAX,
      1 NWIDTH
       write(LLM,90) TITLE,NFNTOT,MAXNOD,DEFMAX,NDMAX,NWIDTH
 90    FORMAT(A,2I8,E16.9,2I8)
 95    FORMAT(20A1,2I8,E16.9,2I8)
       WRITE(LLN) (NSUBT1(I),I=1,80)
       WRITE(LLN) (NSUBT2(I),I=1,80)
       WRITE(LLM,91)
 91    FORMAT('PINE')
 99    FORMAT(80A1)
       LLID=0
       NCOUN=0
C
C           SECOND LOOP TO PICK UP DEFLECTION AT EACH NODE (OF PATRAN MODEL)
C
       DO 400 J=1,KROW
            NCOUN=NCOUN+IA(JNNI+J-1)
            LLPT=IA(JNNI+J)-2
C
C           TAKE OUT RIGID BODY MOTION, RBX,RBZ DISPLACEMENTS AT END BLOCKS
C           ASUMED TO REPRESENT RIGID BODY MOTIONS
```

```
C
              RBY=DEFL(3*(NCOUN+2)-1)
              RBZ=DEFL(3*(NCOUN+2))
              IF (LIQUO.EQ.0) THEN
              RBY=0.
              RBZ=0.
              ENDIF
C IF NO FLUID OPTION DO NOT SUBTACT RIGID BODY CONTRIBUTION
              DO 300 L=1,LLPT
              LLID=LLID+1
              NUMR=NCOUN+L+1
              DX=DEFL(3*NUMR-2)
              DY=DEFL(3*NUMR-1)-RBY
              DZ=DEFL(3*NUMR)-RBZ
              DZLOC=DZ
              VELZ=VELO(3*NUMR)
                IF((A(JVELO-1).NE.0.).OR.(A(JVELO-2).NE.0.)) THEN
                CALL CHCOORD(NUMR,DX,DY,DZ)
                ENDIF
              WRITE(LLN) LLID,DX,DY,DZ,DZLOC,VELZ
              WRITE(LLM,92) LLID,DX,DY,DZ,DZLOC,VELZ
 300          CONTINUE
 400       CONTINUE
C CLOSE FILE OPENED ON UNIT LLN
       CALL CLOSE(LLN)
 92    FORMAT(I8,5E16.9)
       RETURN
       END
C
C
C
       SUBROUTINE CHCOORD(N,X,Y,Z)                                          REPORT 2
       DIMENSION A(1)                                                       BLANC  2
       COMMON IA(1)                                                         BLANC  3
       EQUIVALENCE (IA(1), A(1) )                                           BLANC  4
       COMMON /CPARA/ NSTEPS , NBEG , NEND , NSHEET , N2BD , N3BD1 ,        CPARA  2
      1   N3BD2, INTRVL, DELT, NHTOT, NJOIN, NRELAX, ALPHA                  CPARA  3
      $ LEN SBX                                                             MSPARA 3
       COMMON / SSIZE / IBG(1),NQJ,NELTOT,N1BD,NLOAD,NBRECT,                SSIZE  2
      1   NBQUAD, ISHEET, NPRPTS, NSPTS, NSTRPTS, NVPTS, NHPTS,            SSIZE  3
      2   NSSTYP, NNJ, NNTOT, LGDSP, LIQUD, LBCALC                         SSIZE  4
C                                                                          REPORT 8
       COMMON /STAB / IBT(1),JSSIZE,JSPAR,JVELO,JSTRE,JXMAS,JIELM,JBMAT,  STAB   2
      1   JL1BD,JLOOP,JPRET,JLHIS,JSTRN,JFORC,JXLOC,JNQI,JNNI,JNQBEG,      STAB   3
      $   JLSIDE,JIELMCL,JSTIF,JDEFL,JFORLG,                               STAB   4
      2   JIFPAR,JFLPAR,JXCOORD,JYCOORD,JDELTAX,JDELTAY,JVMA,JSEFX,        STAB   5
      3   JFLUFR, JPRINC, JVELRAD, JGENFR,JPRES,JCSEP                      STAB   6
       COMMON/CIO/ NIN,NOUT,NTHIST,NCORT,MCORT,NTPLOT,NVMA,ITITLE(20)
       COMMON/FRANK/NFNTOT,LLU,LLN,LLS,NOPLTS,NSPLTS
       DIMENSION XMAT(3,3)
       DO 10 I=1,3
       DO 10 J=1,3
       XMAT(I,J)=0.
  10   CONTINUE
C
C              THIS ROUTINE CHANGES THE DISPL. OF NODAL PT. N(FOR EPSA)
C              IN A GLOBAL RECTANGULAR SYSTEM
C
C               IF CURVATURE IN Y DIR. IS NON ZERO ,CALCULATE
C               THE ROTATION MATRIX AT EACH POINT
C
```

```
      IF (A(JVELO-1).NE.0.) THEN
         RCOURY=1./A(JVELO-1)
         THETA1=A(JXLOC+2*N-1)/RCOURY
         XMAT(1,1)=1.
         XMAT(2,2)=COS(THETA1)
         XMAT(2,3)=SIN(THETA1)
         XMAT(3,3)=XMAT(2,2)
         XMAT(3,2)=-1.*XMAT(2,3)
         CALL PROD(XMAT,X,Y,Z)
       ENDIF
C
C            IF CURVATURE IN X DIRECTION IS NON ZERO:
C
       IF (A(JVELO-2).NE.0.) THEN
         DO 20 I=1,3
         DO 20 J=1,3
         XMAT(I,J)=0.
 20    CONTINUE
         RCOURX=1./A(JVELO-2)
         THETA2=A(JXLOC+2*N-2)/RCOURX
         XMAT(2,2)=1.
         XMAT(1,1)=COS(THETA2)
         XMAT(3,3)=XMAT(1,1)
         XMAT(3,1)=-1.*SIN(THETA2)
         XMAT(1,3)=SIN(THETA2)
         CALL PROD(XMAT,X,Y,Z)
       ENDIF
       RETURN
       END
C
C
       SUBROUTINE PROD(XMAT,X,Y,Z)
       DIMENSION XMAT(3,3)
C
C                  THIS ROUTINE DOES THE MATRIX PRODUCT XMAT*(X,Y,Z)
C
       X1=X
       Y1=Y
       Z1=Z
       X=XMAT(1,1)*X1+XMAT(1,2)*Y1+XMAT(1,3)*Z1
       Y=XMAT(2,1)*X1+XMAT(2,2)*Y1+XMAT(2,3)*Z1
       Z=XMAT(3,1)*X1+XMAT(3,2)*Y1+XMAT(3,3)*Z1
       RETURN
       END
C
C
      SUBROUTINE PLOTDISP(DEFL,VELO)
      DIMENSION DEFL(1),VELO(1)
      COMMON/CDELT/ISTEP,TIME
      COMMON /CPARA/ NSTEPS , NBEG , NEND , NSHEET , N2BD , N3BD1 ,      CPARA  2
     1   N3BD2, INTRVL, DELT, NHTOT, NJOIN, NRELAX, ALPHA                CPARA  3
     $ LEN SBx                                                           MSPARA 3
      COMMON/FRANK/NFNTOT,LLU,LLN,LLS,NDPLTS,NSPLTS
C
C          CHECKS IF OUTPUT FOR PATRAN IS REQUESTED BY USER IF YES CALL
C          NEUDISP
C
      IF (TIME.EQ.DELT) KDISP=1
      TIMEI=FLOAT(NSTEPS)/FLOAT(NDPLTS)
      TIMEC=KDISP*TIMEI
      ITIME=JNINT(TIMEC)
```

```
      TIMEC=TIME/(ITIME*DELT)
      ENOTIM=NSTEPS*DELT
      TEST=ABS(TIMEC-JNINT(TIMEC))
      IF((TEST.LE.0.00001).OR.(TIME.EQ.ENOTIM)) THEN
      CALL NEUDISP(OEFL,VELO)
      KOISP=KDISP+1
      ENDIF
      RETURN
      ENO
C
      SUBROUTINE PLOTSTRE(STRE)
      DIMENSION STRE(1)
      COMMON/COELT/ISTEP,TIME
      COMMON /CPARA/ NSTEPS , NBEG , NEND , NSHEET , N2BO , N3BD1 ,      CPARA  2
     1   N3BD2, INTRVL, OELT, NHTOT, NJOIN, NRELAX, ALPHA                CPARA  3
     $ LEN SBX                                                           MSPARA 3
      COMMON/FRANK/NFNTOT,LLU,LLN,LLS,NOPLTS,NSPLTS
C
C        CHECKS IF OUTPUT FOR PATRAN IS REQUESTED BY USER IF YES CALL
C        NEUDISP
C
      IF (TIME.EQ.OELT) KSTRE=1
      TIMEI=FLOAT(NSTEPS)/FLOAT(NSPLTS)
      TIMEC=KSTRE*TIMEI
      ITIME=JNINT(TIMEC)
      TIMEC=TIME/(ITIME*OELT)
      ENDTIM=NSTEPS*DELT
      TEST=ABS(TIMEC-JNINT(TIMEC))
      IF((TEST.LE.0.00001).OR.(TIME.EQ.ENOTIM)) THEN
      CALL NEUSTRE(STRE)
      KSTRE=KSTRE+1
      ENOIF
      RETURN
      END
C
C
      SUBROUTINE NEUSTRE(STRE)                                           REPORT 2
      OIMENSION A(1)                                                     BLANC  2
      COMMON IA(1)                                                       BLANC  3
      EQUIVALENCE (IA(1), A(1) )                                         BLANC  4
      OIMENSION OISP(10),STRA(10),DSTRE(10),OS(5),VEL(2)
      OIMENSION STRE(1)
      COMMON /CPARA/ NSTEPS , NBEG , NEND , NSHEET , N2BO , N3BO1 ,      CPARA  2
     1   N3BO2, INTRVL, DELT, NHTOT, NJOIN, NRELAX, ALPHA                CPARA  3
     $ LEN SBX                                                           MSPARA 3
      COMMON / SSIZE / IBG(1),NQJ,NELTOT,N1BO,NLOAO,NBRECT,              SSIZE  2
     1   NBQUAO, ISHEET, NPRPTS, NSPTS, NSTRPTS, NVPTS, NHPTS,          SSIZE  3
     2   NSSTYP, NNJ, NNTOT, LGOSP, LIQUO, LBCALC                       SSIZE  4
C                                                                       REPORT 8
      COMMON /STAB / IBT(1),JSSIZE,JSPAR,JVELO,JSTRE,JXMAS,JIELM,JBMAT, STAB   2
     1   JL1BD,JLOOP,JPRET,JLHIS,JSTRN,JFORC,JXLOC,JNQI,JNNI,JNQBEG,     STAB   3
     $   JLSIOE,JIELMCL,JSTIF,JOEFL,JFORLG,                             STAB   4
     2   JIFPAR,JFLPAR,JXCOORO,JYCOORD,JOELTAX,JOELTAY,JVMA,JSEFX,       STAB   5
     3   JFLUFR, JPRINC, JVELRAO, JGENFR,JPRES,JCSEP                     STAB   6
      COMMON/CIO/ NIN,NOUT,NTHIST,NCORT,MCORT,NTPLOT,NVMA,ITITLE(20)
      COMMON/FRANK/NFNTOT,LLU,LLN,LLS,NOPLTS,NSPLTS
      COMMON/TITRE/NTITLE(80)
      COMMON/COELT/ISTEP,TIME
```

```
      COMMON/SPARA/REG(1),E,GNU,RHO,SGYLD,THICK,CURV(2)
      INTEGER  NSUBT1(80),NSUBT2(80)
      NWIDTH=31
      LLS=16
      LLZ=15
         DO 10 J=1,80
 10      NSUBT2(J)=0
C TAKE CARE OF THE 2ND TITLE , INDICAT. TIME STEP
C
      CLOSE(UNIT=LLZ)
      OPEN(UNIT=LLZ,STATUS='NEW')
      WRITE(LLZ,*) TIME
      CLOSE(UNIT=LLZ)
      READ(LLZ,801) (NSUBT1(I),I=1,80)
      CLOSE(UNIT=LLZ)
      OPEN(UNIT=LLZ,STATUS='OLD')
 801  FORMAT(80A1)
      OPEN(UNIT=LLS,FORM='UNFORMATTED',STATUS='NEW')
C
C
C WE ARE GOING TO DISPLAY THE FIRST THREE RECORDS (TITLES)
C
      WRITE(LLS) (NTITLE(I),I=1,80),NWIDTH
      WRITE(LLS) (NSUBT1(I),I=1,80)
      WRITE(LLS) (NSUBT2(I),I=1,80)
C
C        WE ARE GOING TO SCAN ALL ELEMENTS AS IN SHCONN,
C        GET THE STRESSES PERFORM ROTATION IF NECESSARY
C        LLEL: NUM. OF ROWS ELTS; LLID:ELT NUMBER
C        NPREV: NODE NUM.(FOR EPSA) OF LAST POINT OF ROW JUST
C        BELOW THE ELEMENT ROW K; LLROW: NUM. OF ELTS IN EACH ROW
C
      NPREV=0
C              IA ELTS +1(FOR NODES) +2 ARTIFICIAL NODES FOR NPREV
      NSHAPE=4
      LLEL=JNNI-JNQI
        DO 200 K=1,LLEL
C
        LLROW=IA(JNQI+K-1)
        NPREV=NPREV+LLROW+3
C
             DO 100 J=1,LLROW
             LLID=IA(JNQBEG+K-1)+J-1
             DS(1)=STRE(9*(LLID-1)+1)
             DS(2)=STRE(9*(LLID-1)+2)
             DS(3)=0.
C DS(4,5) WILL BE THE STRESSES IN LOCAL COORD.
             DS(4)=DS(1)
             DS(5)=DS(2)
C VON MISES AT TOP AND BOTTOM OF SHELL:
             TOXY=STRE(9*(LLID-1)+3)
             DSX=STRE(9*(LLID-1)+4)
             DSY=STRE(9*(LLID-1)+5)
             DSX=DSX*6./THICK
             DSY=DSY*6./THICK
             DSX1=DS(1)+DSX
             DSY1=DS(2)+DSY
             DSX2=DS(1)-DSX
             DSY2=DS(2)-DSY
C
C  VON MISES CALCULATIONS MODIFIED AFTERWARDS FOR MEMB.+BENDING STRESS
```

```
C  NO SHEAR AT TOP OF SHELL,GET STRESSES AT TOP AND BOTTOM AND TAKE
C  MAX. VALUE
C
C                                  TOP:
                SIGM1=OSX1
                SIGM2=OSY1
                VONM1=SQRT(SIGM1*SIGM1-SIGM1*SIGM2+SIGM2*SIGM2)
C                                  BOTTOM:
                SIGM1=OSX2
                SIGM2=OSY2
                VONM2=SQRT(SIGM1*SIGM1-SIGM1*SIGM2+SIGM2*SIGM2)
C
                VONM=AMAX1(VONM1,VONM2)
C     O.K. FOR VON MISES (VONM)
C     IS THE SHELL CURVED, NLELI'S ARE NODES SURROUNDING ELT. (EPSA)
C
                 IF((A(JVELO-1).NE.0.).OR.(A(JVELO-2).NE.0.)) THEN
                 NLEL1=J+NPREV+1
                 NLEL2=J+NPREV+2
                 NLEL3=J+1+NPREV+LLROW+1+2
                 NLEL4=J+NPREV+LLROW+1+2
                 X=(A(JXLOC+2*NLEL1-2)+A(JXLOC+2*NLEL2-2))/4.
                 X=X+(A(JXLOC+2*NLEL3-2)+A(JXLOC+2*NLEL4-2))/4.
                 Y=(A(JXLOC+2*NLEL1-1)+A(JXLOC+2*NLEL2-1))/4.
                 Y=Y+(A(JXLOC+2*NLEL3-1)+A(JXLOC+2*NLEL4-1))/4.
                 CALL CHELEM(X,Y,OS)
                 ENDIF
C
C      READY TO DISPLAY RECORD
C
               WRITE(LLS) LLID,NSHAPE,(DISP(I),I=1,10),
     1            (STRA(I),I=1,10),DSTRE(1),DS(5),DS(6),
     2             DSTRE(4),(DS(I),I=1,3),
     3            (DSTRE(I),I=8,10),VONM
C              WRITE(15,80) LLID,NSHAPE,(DS(I),I=1,5),VONM
  100      CONTINUE
  200   CONTINUE
        CALL CLOSE(LLS)
   80   FORMAT(2I4,6E15.8)
        RETURN
        END


        SUBROUTINE CHELEM(X,Y,OS)                                          REPORT 2
        DIMENSION A(1)                                                     BLANC  2
        COMMON IA(1)                                                       BLANC  3
        EQUIVALENCE (IA(1), A(1) )                                         BLANC  4
        COMMON /CPARA/ NSTEPS , NBEG , NEND , NSHEET , N2BD , N3BD1 ,      CPARA  2
     1    N3BD2, INTRVL, DELT, NHTOT, NJOIN, NRELAX, ALPHA                 CPARA  3
     $ LEN SBX                                                             MSPARA 3
        COMMON / SSIZE / IBG(1),NQJ,NELTOT,N1BD,NLOAD,NBRECT,              SSIZE  2
     1    NBQUAD, ISHEET, NPRPTS, NSPTS, NSTRPTS, NVPTS, NHPTS,           SSIZE  3
     2    NSSTYP, NNJ, NNTOT, LGOSP, LIQUD, LBCALC                        SSIZE  4
C                                                                          REPORT 8
        COMMON /STAB / IBT(1),JSSIZE,JSPAR,JVELO,JSTRE,JXMAS,JIELM,JBMAT, STAB   2
     1    JL1BD,JLOOP,JPRET,JLHIS,JSTRN,JFORC,JXLOC,JNQI,JNNI,JNQBEG,     STAB   3
     $    JLSIDE,JIELMCL,JSTIF,JDEFL,JFORLG,                              STAB   4
     2    JIFPAR,JFLPAR,JXCOORD,JYCOORD,JDELTAX,JDELTAY,JVMA,JSEFX,       STAB   5
     3    JFLUFR, JPRINC, JVELRAD, JGENFR,JPRES,JCSEP                     STAB   6
        DIMENSION XMAT(3,3)
        DIMENSION DS(3)
```

```
      DO 10 I=1,3
      DO 10 J=1,3
      XMAT(I,J)=0.
   10 CONTINUE
C
C THIS ROUTINE CHANGES THE STRESSE IN AN ELEMENT INTO A
C IN A GLOBAL RECTANGULAR SYSTEM ,GIVEN LOCAL COORD. OF
C CENTROID X,Y
C
C            IF CURVATURE IN Y DIR. IS NON ZERO ,CALCULATE
C    THE ROTATION MATRIX AT EACH POINT
      IF (A(JVELO-1).NE.0.) THEN
         RCOURY=1./A(JVELO-1)
         THETA1=Y/RCOURY
         XMAT(1,1)=1.
         XMAT(2,2)=COS(THETA1)
         XMAT(2,3)=SIN(THETA1)
         XMAT(3,3)=XMAT(2,2)
         XMAT(3,2)=-1.*XMAT(2,3)
         CALL PROD(XMAT,DS(1),DS(2),DS(3))
       ENDIF
C
C IF CURVATURE IN X DIRECTION IS NON ZERO:
       IF (A(JVELO-2).NE.0.) THEN
         DO 20 I=1,3
         DO 20 J=1,3
         XMAT(I,J)=0.
  20     CONTINUE
         RCOURX=1./A(JVELO-2)
         THETA2=X/RCOURX
         XMAT(2,2)=1.
         XMAT(1,1)=COS(THETA2)
         XMAT(3,3)=XMAT(1,1)
         XMAT(3,1)=-1.*SIN(THETA2)
         XMAT(1,3)=SIN(THETA2)
         CALL PROD(XMAT,DS(1),DS(2),DS(3))
       ENDIF
       RETURN
       END
```

LIST OF REFERENCES

1. Dynamic Elasto Plastic Response of Shells in an Acoustic Medium User's Manual for the EPSA Code, Weidlinger Asscciates, New York, NY, March 1980

2. Newton, R. E. , Zienkiewicz, O. C. Coupled Vibrations of a Structure Submerged in a Compressible Fluid , Int. Symposium on Finite Element Techniques , Stuttgart, June 1969

3. Geers, T. L., "Residual Potential and Approximate Methods for Three Dimensional Fluid-Structure Interaction Problems" , J. Acoustic Soc. Am. , Vol 49, No. 5, May 1971, pp 1505-1510

4. Atkatsh, R. ,Bienek ,M.P.,Baron, M.L. Dynamic Elasto Plastic Response of Shells in an Acoustic Medium. Theoretical Development for the EPSA code , Weidlinger Associates, New York, NY, July 1978

5. Elasto Plastic Behavior of Plates and Shells, Weidlinger Associates, New York, NY , March 1976

6. Bathe, K. J. Finite Element Procedures in Engineering Analysis , Prentice Hall, January 1982

7. PATRAN-G user's guide, PDA Engineering, Santa Ana, CA, 1980

8. Timoshenko, S. , Woinowsky-Krieger, S. Theory of Plates and Shells , Mc Graw-Hill , New York, NY, 1959

9. Kinsler, L.E. ,Frey, A. R. Fundamentals of Acoustics, John Wiley and sons , New York, N.Y. ,1962

10. Cole, R. H. Underwater Explosions , Princeton University Press , Princeton, NJ, 1948

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California, 93943	2
3.	Professor Y.S. Shin, Code 69Sb Department of Mechanical Engineering Naval Postgraduate School Monterey, California, 93943	3
4.	Professor G. Cantin, Code 69 Ci Department of Mechanical Engineering Naval Postgraduate School Monterey, California, 93943	1
5.	Department Chairman, Code 69 Department of Mechanical Engineering Naval Postgraduate School Monterey, California, 93943	1
6.	Georges Daube, Senior Engineer 25/27, Blvd Arago 75013 Paris, FRANCE	2
7.	French Scientific Mission 2011, Eye Street Washington, D.C. 20019	1
8.	R. Daddazio, Weidlinger Associates 333, Seventh Avenue New York, NY, 10001	1
9.	M.L. Baron, Weidlinger Associates 333, Seventh Avenue New York, NY, 10001	1
10.	R. Atkatsh, Weidlinger Associates 333, Seventh Avenue New York, NY, 10001	1
11.	T.L. Geers, Lockeed Missile and Space Company 3251 Hanover Street Palo Alto, CA 94304	1
12.	J. Gagorik, code 03R24 NAVSEA Washington, DC 20362	1
13	Lt. H. Reams, SSPS Defense Nuclear Agency Washington, DC, 20305	1

14. B. Whang, code 1750.2 — 1
Hull group head, Submarine Protection Div.
David Taylor Naval Ship Research and
Development Center
Bethesda, MD, 20084

15. Mr C. Cochard, Mr. D. Bergeze SNEA(P) — 1
DRMTA, Chemin Vignancour
64000 Pau FRANCE

16. Mr L. Barbouteau — 1
Elf-Aquitaine, Les Miroirs
92 Paris La Defense FRANCE

17. Daniel Finifter — 1
47, rue de Monceau
75008 Paris, FRANCE

206174

Thesis
D1619 Daube
c.1 Underwater shock-induced responses of submerged cylindrical structures.